



# PREMIS Introduction

**Karin Bredenberg**  
National Archives of Sweden

**Angela Di Iorio**  
Sapienza Università di Roma

**Eld Zierau**  
Royal Danish Library

**Bertrand Caron**  
Bibliothèque nationale de France

## Purpose of the Tutorial

- Introduce PREMIS
- Set a baseline of common understanding
- Introduce you to the PREMIS community
- Answer questions:
  - We might need to put them in parking but all questions will be answered!

## Who are we?

From the PREMIS Editorial Committee:

- Angela
- Bertrand
- Eld
- Karin

Form the audience via show of hands:

- Have you heard of PREMIS?
- Do you know the PREMIS data model?
- Have you used PREMIS in practice?
- Have you modelled environments?
- Are you from
  - a library?
  - an archive?
  - a university?
  - Something else?



## Agenda

### 9:30-10:10 **Introduction to PREMIS**

Welcome

Background (brief history and rationale)

Benefits of implementing PREMIS

Outline of main Entities

### 10:10-10:50 **Implementation community support tools**

Data Dictionary

Ontology & Different expressions of PREMIS DD

PREMIS Conformance & interoperability

### 10:50-11:10 **Implementation case studies**

PREMIS in METS

Strategies for implementing PREMIS with Semantic Web Technology

### 11:10-11:25 **Wrap Up**

Website, PIG, id.loc.gov

Exercise (Objects, Events, Agents, Rights)

**Karin Bredenberg**

The National Archives of Sweden



# **DIGITAL PRESERVATION METADATA - WHY IS IT NEEDED AND WHAT DOES IT LOOK LIKE?**

## What is digital preservation metadata?

- Digital preservation metadata =  
Metadata to ensure long-term accessibility  
of digital resources
- Digital objects must be self-descriptive
- Must be able to describe, manage and discover  
independently from the systems that were used to  
create them  
XML (machine and human readable)
- Often bundled with the content files  
in an information package

## Domain

Born digital



Digitized



## DP metadata supports preservation goals



**Karin Bredenberg**

The National Archives of Sweden



# WHAT IS PREMIS?



## The PREMIS standard

- International *de-facto* standard for metadata to support the preservation of digital objects and ensure their long-term usability.
  - Information you need to know for preserving digital objects
    - Preservation Metadata: Implementation Strategies*
- Developed by an international team of experts.
- Implemented in digital preservation projects around the world.
- Incorporated into commercial and open-source digital preservation tools and systems.

# The PREMIS standard

- Data Dictionary (PREMIS 3.0)
  - <http://www.loc.gov/standards/premis/v3/premis-3-0-final.pdf>
  - Version 3 – major release
- XML schema v3.0
  - <http://www.loc.gov/standards/premis/premis.xsd>
- OWL ontology
- Supporting documentation



## Activities

- The PREMIS Editorial Committee
  - Coordinates revisions and implementation of the standard
- PREMIS Implementors' Group forum (pig@loc.gov)
  - Email message to [listserv@loc.gov](mailto:listserv@loc.gov):  
Text: subscribe pig <your name>
- Preservation metadata workshop (@iPres)

## Scope

- What PREMIS DD is:
  - Common data model for organizing/thinking about preservation metadata
  - Standard for exchanging preservation metadata in information packages between repositories
  - Implementable
  - Technically neutral
  - Core metadata

## Scope

- What PREMIS DD is not:
  - Out-of-the-box solution
  - All needed metadata
  - Lifecycle management of objects outside repository
    - increasing support for integration with outside
  - Rights management standard
    - strong support for rights statements

## Scope

- What PREMIS DD is not:
  - It is not limited to or customized for archives and libraries.
  - It does not dictate that you need to use every feature.
    - But you should examine for yourself which features you can knowingly ignore.
  - It is not only useful if you implement metadata. You can use it to assess the metadata quality of systems you use.
  - Everyone modeling the digital landscape can and should use the high-level modeling feature.

## Tailoring PREMIS to needs

- Evolving metadata
  - Increasing experience ensuring the longevity of digital objects
  - Changing future technical possibilities
  - Changing future legal framework
- Tailoring solutions
  - Varying needs
    - Content-types
    - Institutional policies
    - Intended use
  - Off-the-shelf (OS/commercial) or custom-built

### Off-the-shelf systems

- Predefined metadata profiles
- Out-of-the-box tools

### Configured, extended, adapted

- Metadata profiles and tools

### Custom-built systems

- Metadata profiles and tools

Angela Di Iorio  
Sapienza Università di Roma



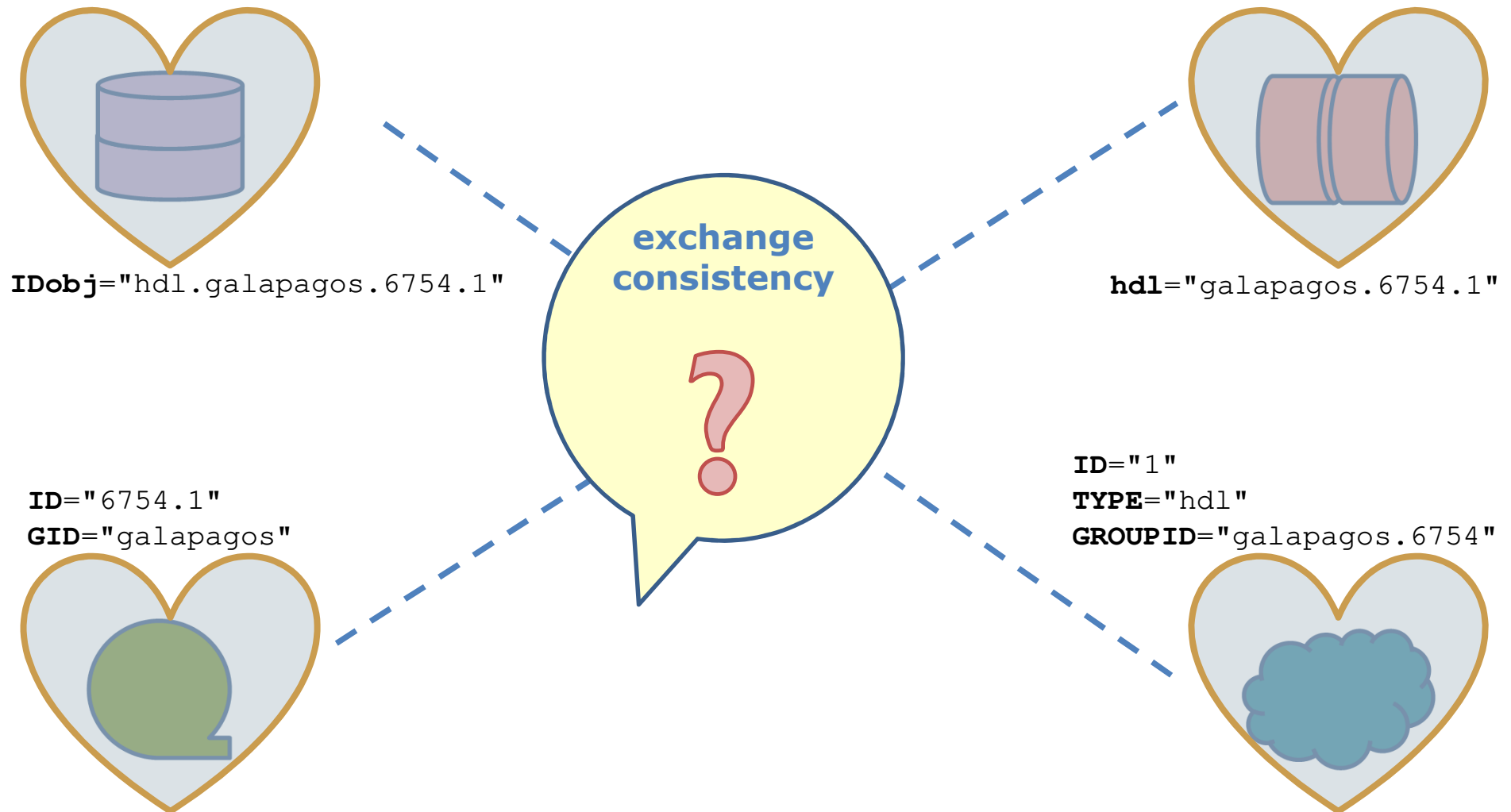
# **BENEFITS OF IMPLEMENTING PREMIS**



## Benefits of implementing PREMIS

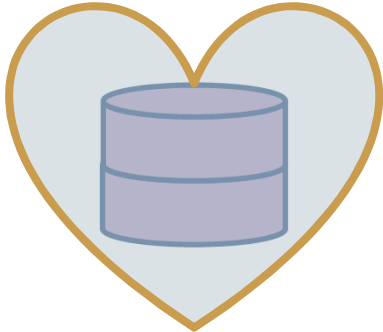
- supports the viability, renderability, understandability, authenticity, and identity of digital objects in a preservation context;
- information most preservation repositories need to know to preserve digital materials over the long term;
- “implementable metadata”: rigorously defined, supported by guidelines for creation, management, and use, and oriented toward automated workflows;
- technical neutrality: no assumptions made about preservation technologies, strategies, metadata storage and management, etc.

## Benefits of implementing PREMIS



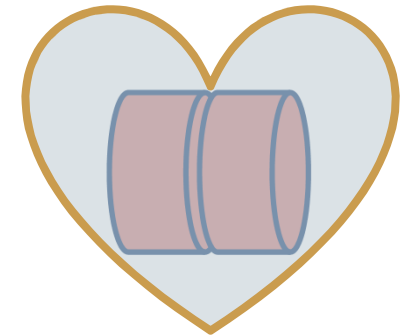
## Benefits of implementing PREMIS

**IDobj**="hdl.galapagos.6754.1"



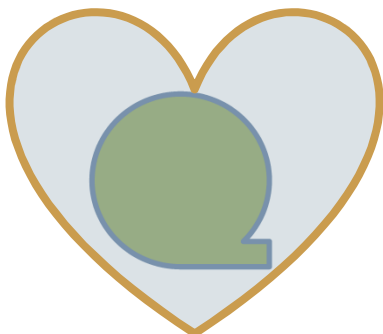
```
premis:objectIdentifier
premis:objectIdentifierType="hdl"
premis:objectIdentifierValue="galapagos.6754.1"
```

**hdl**="galapagos.6754.1"



### PREMIS

```
<galapagos.6754.1> a premisOwl:IntellectualEntity ;
premisOwl:identifier
<http://hdl.handle.net/galapagos.6754.1> .
<http://hdl.handle.net/galapagos.6754.1> a
http://id.loc.gov/vocabulary/identifiers/hdl .
```



**ID**="6754.1"  
**GID**="galapagos"



**ID**="1"  
**TYPE**="hdl"  
**GROUPID**="galapagos.6754"

## Implementable, core preservation metadata

“things that most working preservation repositories are likely to need to know in order to support digital preservation.”

- **CORE and IMPLEMENTABLE**

- any metadata absolutely required under any circumstances
- information that a repository needs to know, regardless of how or whether is stored
- not necessarily mandatory

**Eld Zierau**

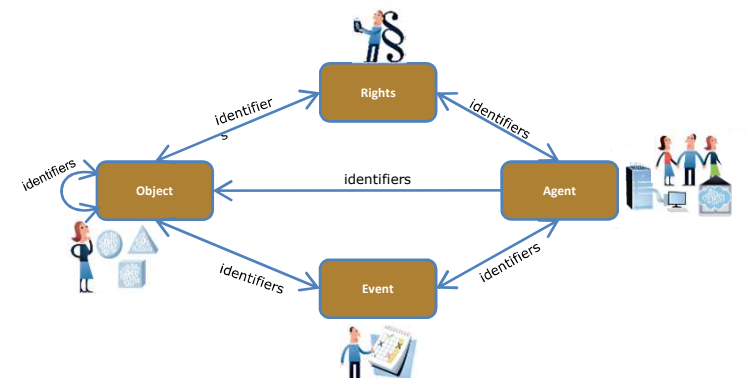
Royal Danish Library

# OUTLINE OF MAIN ENTITIES



## THE DATA MODEL & KEY CONCEPTS

## DATA DICTIONARY DESCRIPTION OF DATA MODEL



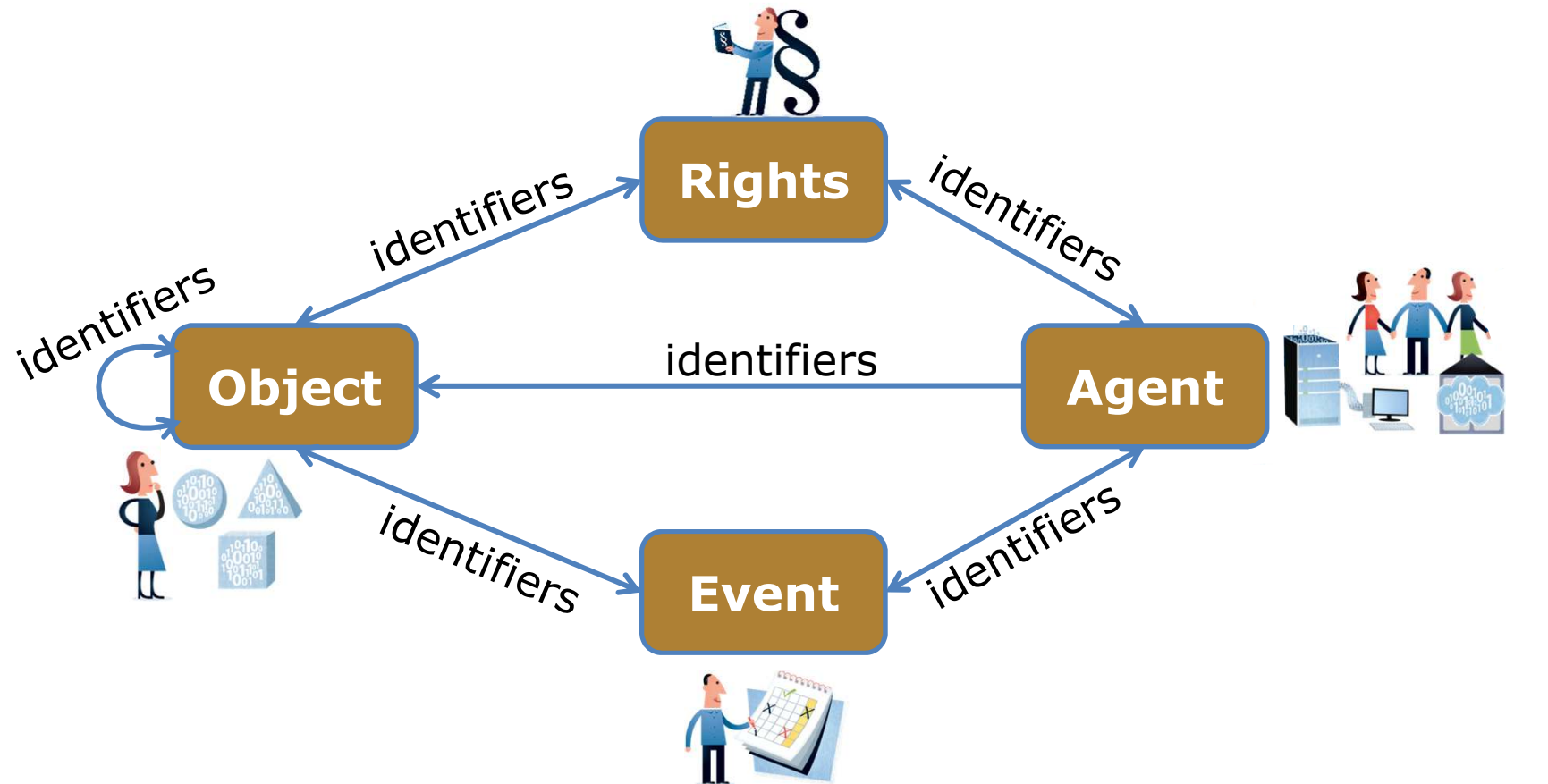


## The PREMIS Data Model



- Data model includes:
  - Entities: “things” relevant to digital preservation that are described by preservation metadata (Intellectual Entities, Objects, Events, Rights, Agents)
  - Properties of Entities (semantic units)
  - Relationships between Entities
- Why have a data model?
  - Organizational convenience (for development and use)
  - Useful framework for distinguishing applicability of semantic units across different types of Entities and different types of Objects
  - But: not a formal entity-relationship model; not sufficient to design databases

## PREMIS 3 - Entities



## PREMIS 3 Entities



### object

Are what repository actually preserves  
Different types of objects: 'file', 'representation', 'bitstream' or 'intellectual entity'



### event

Aggregates metadata about actions.  
Contains e.g. event identifier, event type (creation, migration, ...), and other event details.



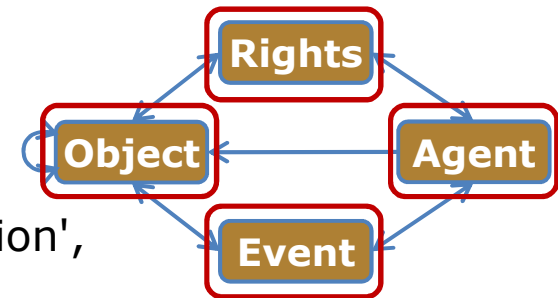
### agent

Can have different roles in relation to an event or objects.  
Can be persons, organisations, programs etc.  
Contains identifier and possibly other information like name and type

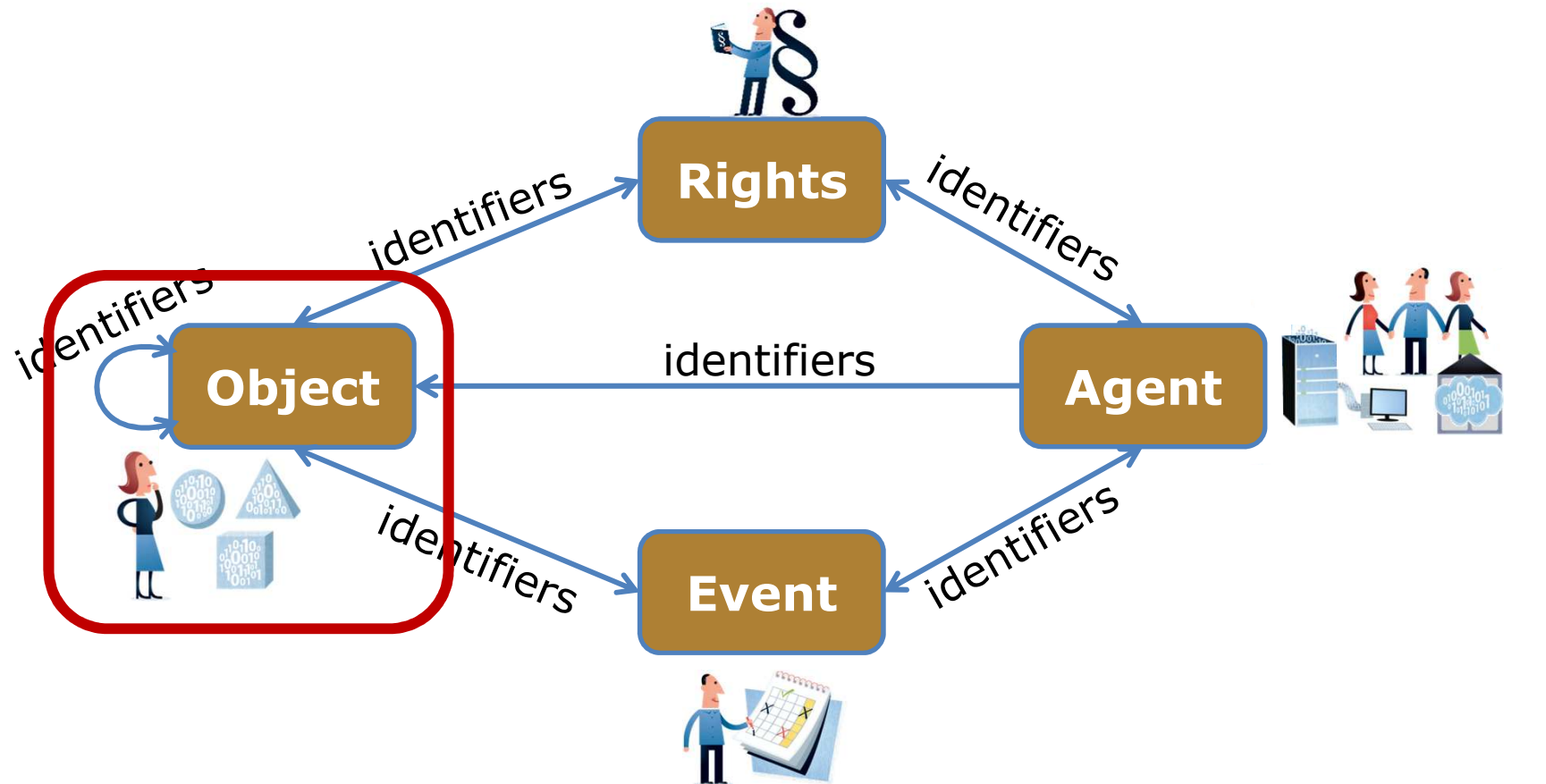


### rights

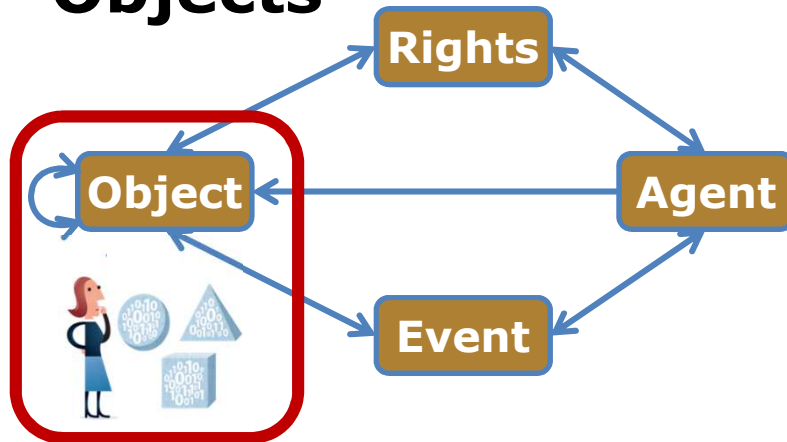
Describes rights and permissions specifically related to digital preservation.  
Contains e.g. rights statements and possible extensions with other metadata



## PREMIS 3 Entities



## Objects

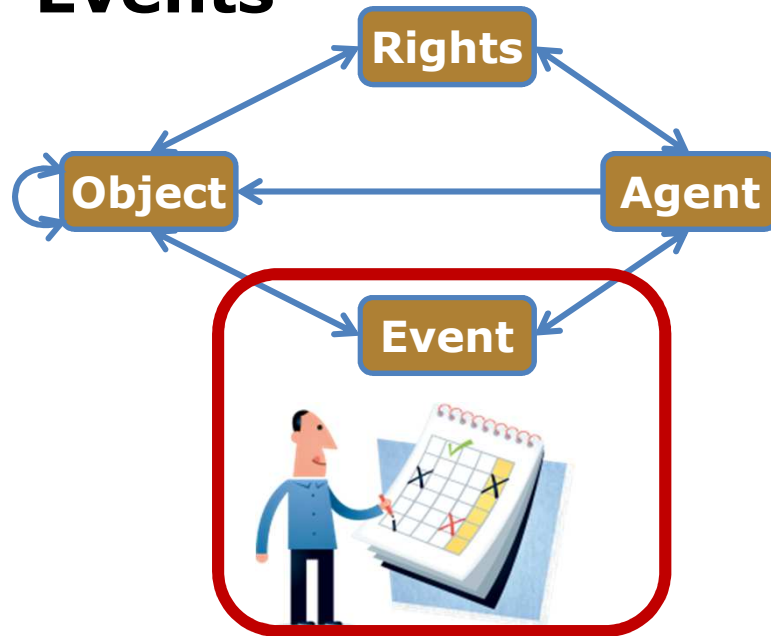


### Examples:

- A book
- A book representation
- A PDF file
- An image within a file

- It is the objects that you preserve
- Objects can be intellectual entities, representations or bitstreams (more later)
- A repository does NOT have to manage all types of Objects
- If a repository has more types you need to record relationships between them

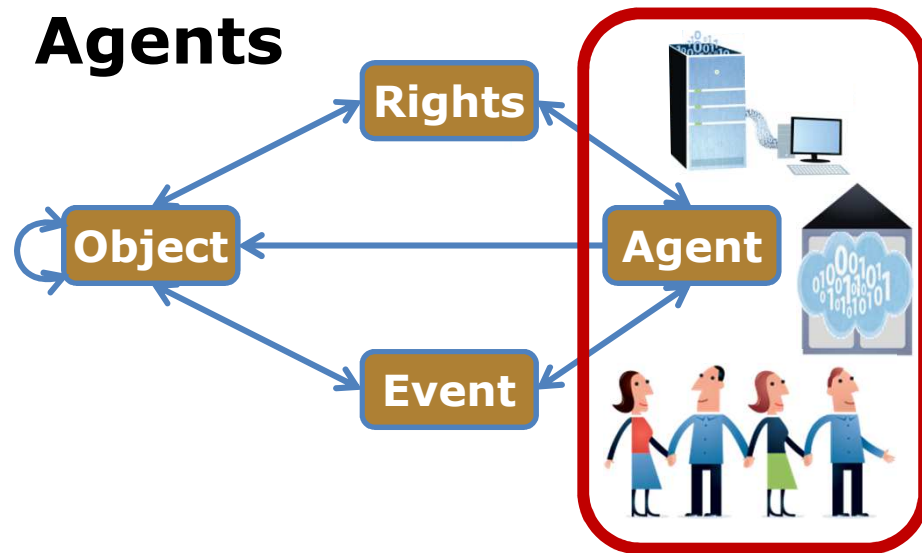
## Events



### Examples:

- Validation Event: use JHOVE tool to verify that chapter1.pdf is a valid PDF file
  - Ingest Event: transform an OAIS SIP into an AIP (one Event or multiple Events?)
- An action that involves or impacts at least one Object or Agent associated with or known by the preservation repository
  - Helps document digital provenance. Can track history of Object through the chain of Events that occur during the Objects lifecycle
  - Determining which Events are in scope is up to the repository (e.g., Events which occur before ingest, or after de-accession)
  - Determining which Events should be recorded, and at what level of granularity is up to the repository

## Agents

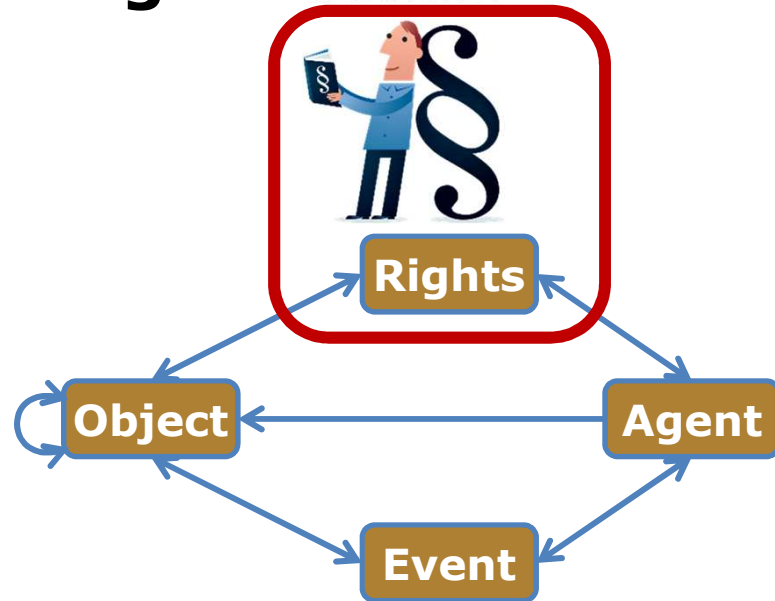


- Person, organization, or software program/system associated with an rights management and/or preservation events in the life of an object
- Intended only to identify the agent unambiguously, and to allow linking from other entity types.
- Repositories are encouraged to use any richer scheme that may be appropriate.

### Examples:

- **John Smith** (a person)
- **IIPC** (an organization)
- **JHOVE version 1.5** (a software program)

## Rights Statements



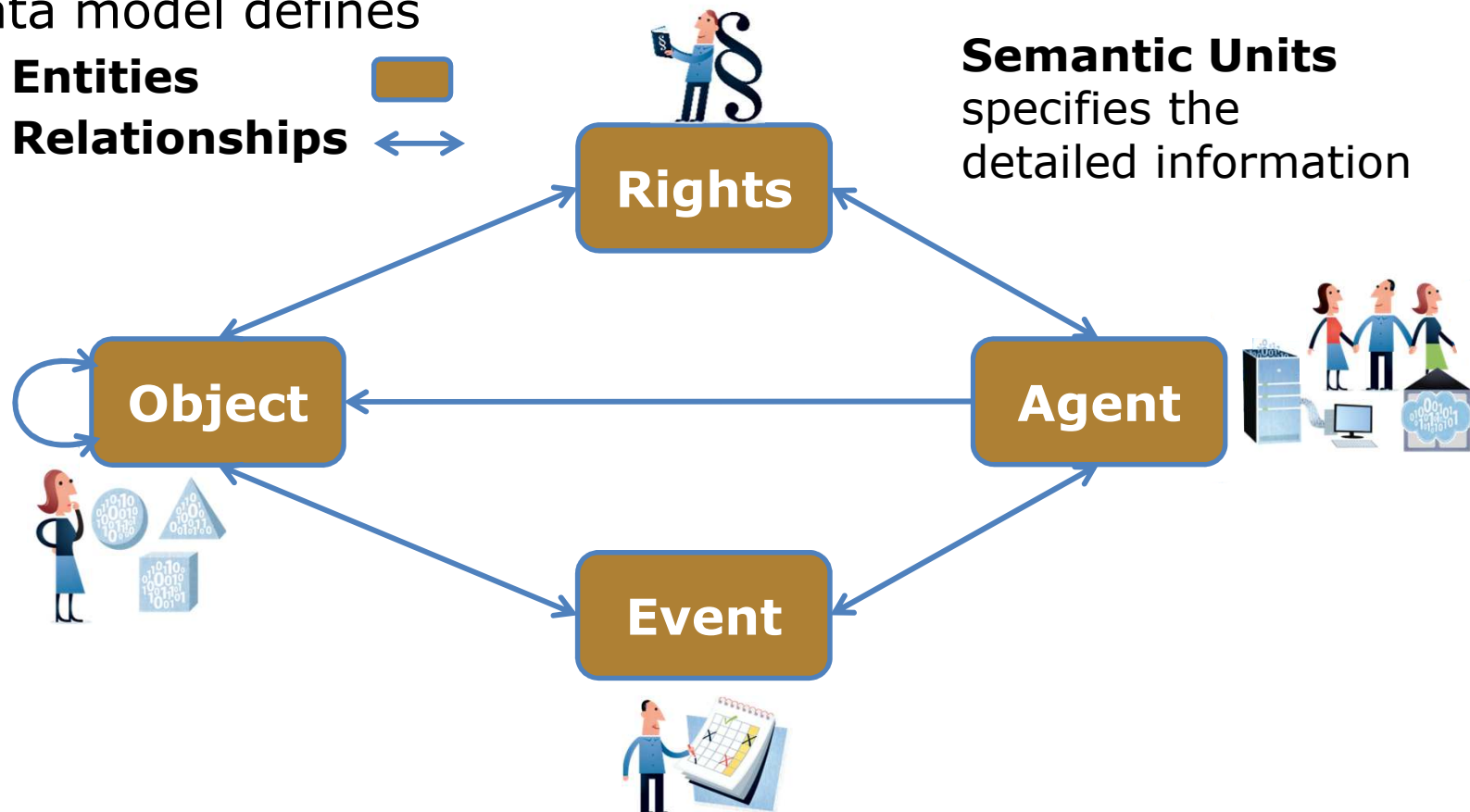
### Example:

- Helen Smith grants FCLA digital repository permission to make three copies of metadata\_fundamentals.pdf for preservation purposes.
- An agreement with a rights holder that grants permission for the repository to undertake an action(s) associated with an Object(s) in the repository.
- Not a full rights expression language; focuses on permissions that take the form:
  - Agent X grants Permission Y to the repository in regard to Object Z.
- Basis for rights may be copyright, license, statute or other e.g. institutional policy

## Properties of Entities - Semantic units

Data model defines

- **Entities** 
- **Relationships** 

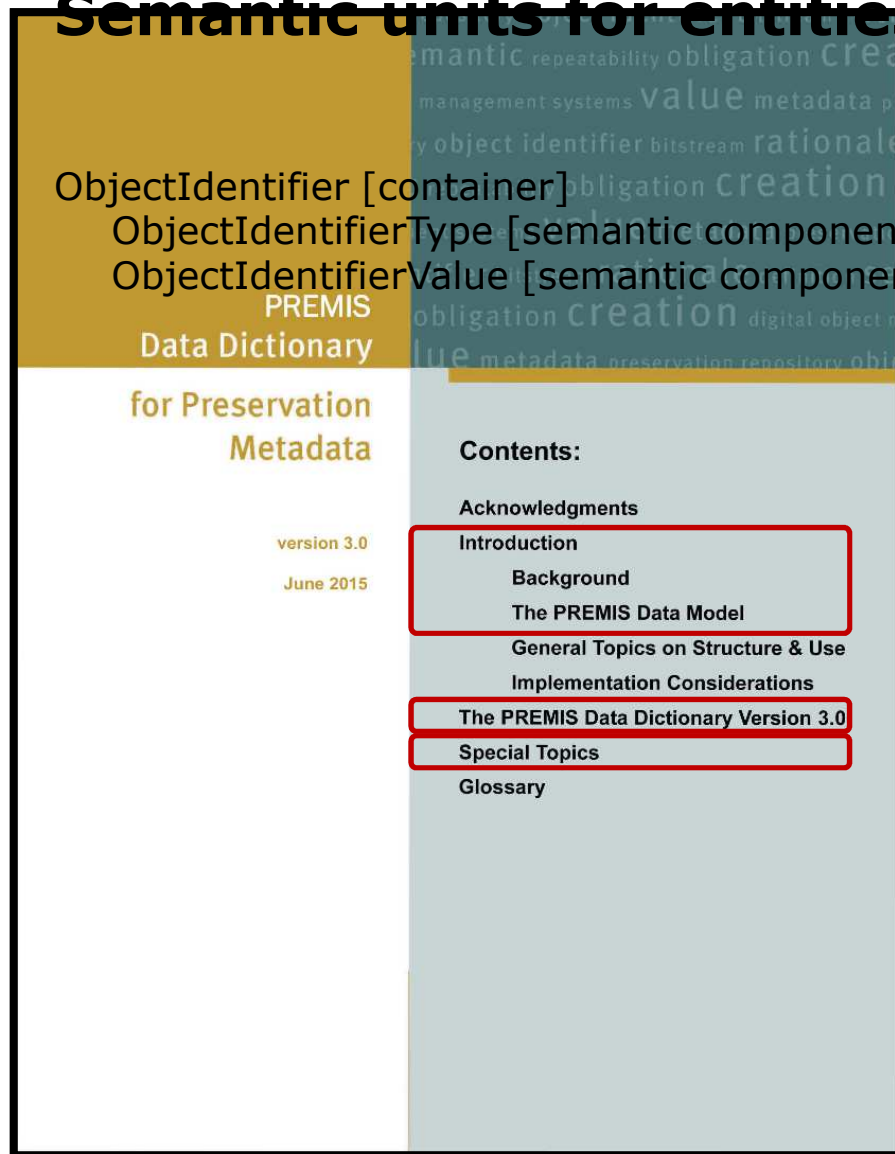


## Semantic Units

- A **semantic unit** is a property of an entity:
  - Something you *need to know* about an entity
  - A piece of information most repositories need to know in order to carry out their digital preservation functions
- Two kinds of semantic unit:
  - **Container**: groups together related semantic units
  - **Semantic components**: semantic units grouped under the same container
- **Example** – a semantic unit for object:
  - ObjectIdentifier [container]
    - ObjectIdentifierType [semantic component]
    - ObjectIdentifierValue [semantic component]

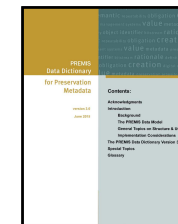
### Semantic units for entities

ObjectIdentifier [container]  
ObjectIdentifierType [semantic component]  
ObjectIdentifierValue [semantic component]



### Data Dictionary (PREMIS 3.0)

<http://www.loc.gov/standards/premis/v3/premis-3-0-final.pdf>



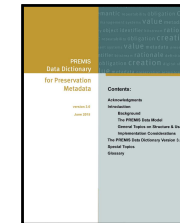
## Semantic units for entities XML example

ObjectIdentifier [container]  
 ObjectIdentifierType [semantic component]  
 ObjectIdentifierValue [semantic component]

```
<premis>
  <object xsi:type="file">
    <objectIdentifier>
      <event>
        <ObjectIdentifierType>
          <agent>
            </agent>
          </ObjectIdentifierType>
          <ObjectIdentifierValue>
            41d10-099-1e2-9
          </ObjectIdentifierValue>
        </ObjectIdentifier>
      </event>
    </object>
  <event> ... </event>
  <agent> ... </agent>
  <rights> ... </rights>
</premis>
```

## Data Dictionary (PREMIS 3.0)

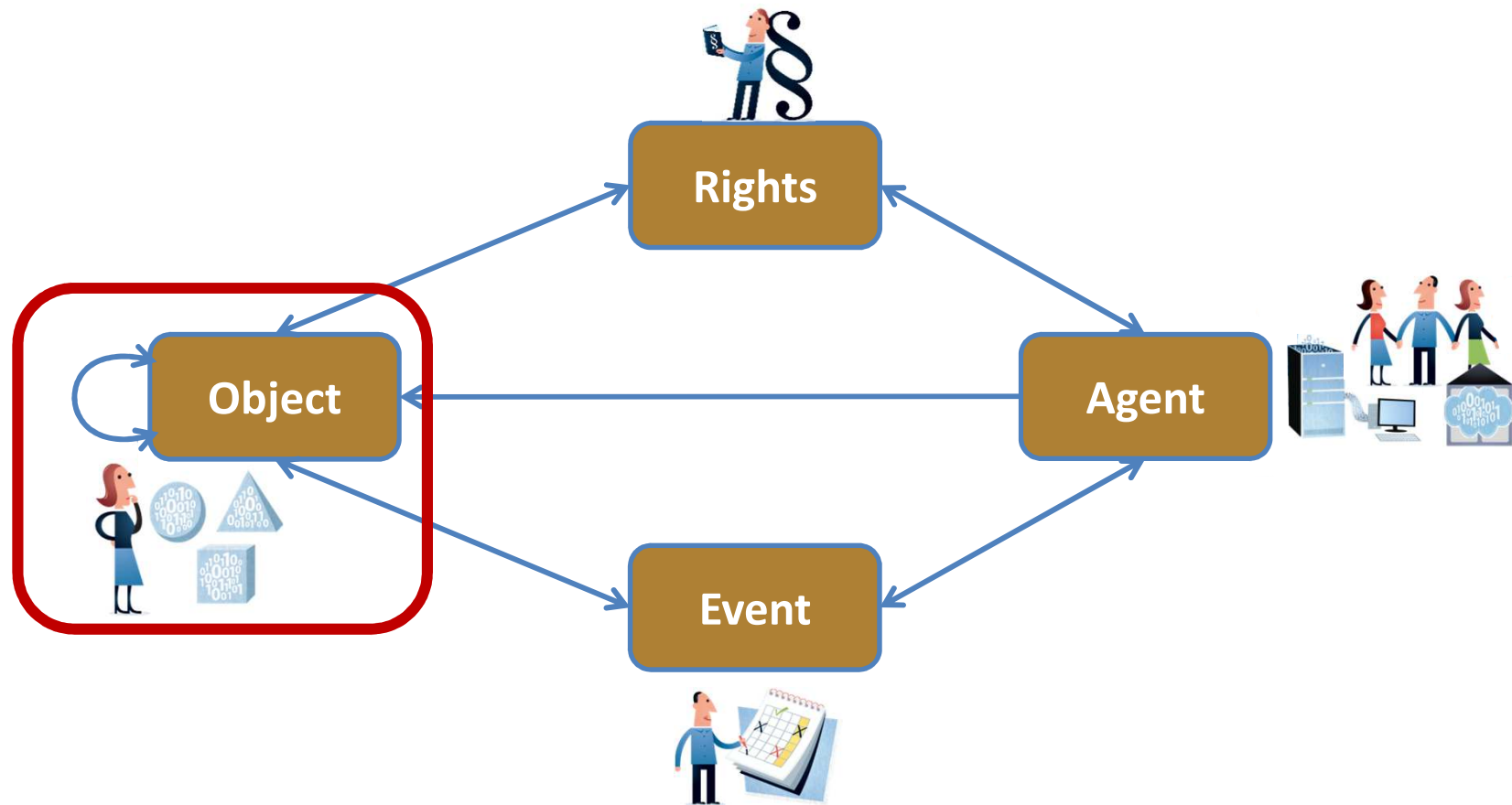
<http://www.loc.gov/standards/premis/v3/premis-3-0-final.pdf>

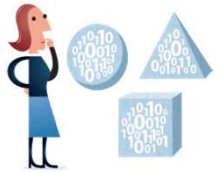


**Exemplified** by XML using  
XML schema v3.0:

<http://www.loc.gov/standards/premis/premis.xsd>

## Properties of Entities - Semantic units





## High level semantic units for Objects

what technical  
information on it?



which object is it?

`ark:/12148/btp6k102002g/f1`

what is my preservation  
strategy for this object?

what kind of object?

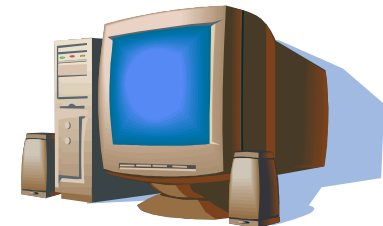


where is it stored?  
on which media?

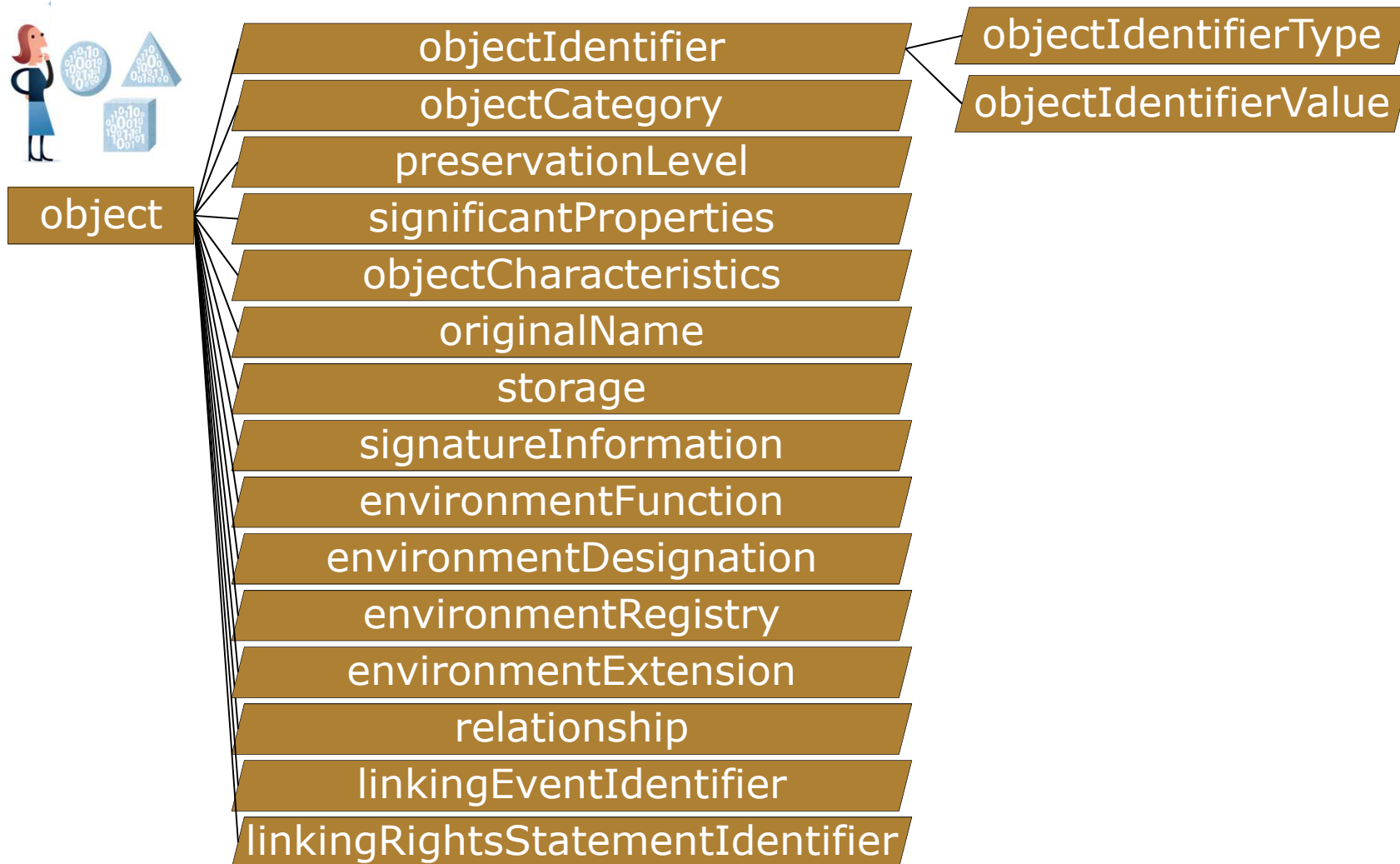


which of its  
characteristics  
do I want to  
preserve in it?

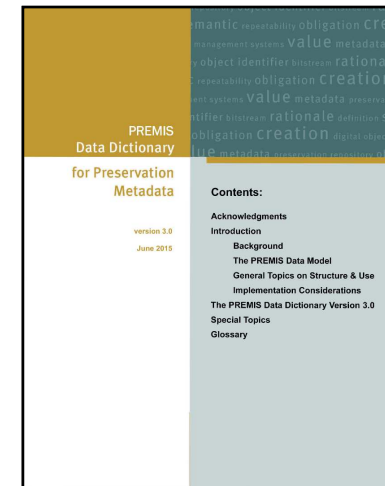
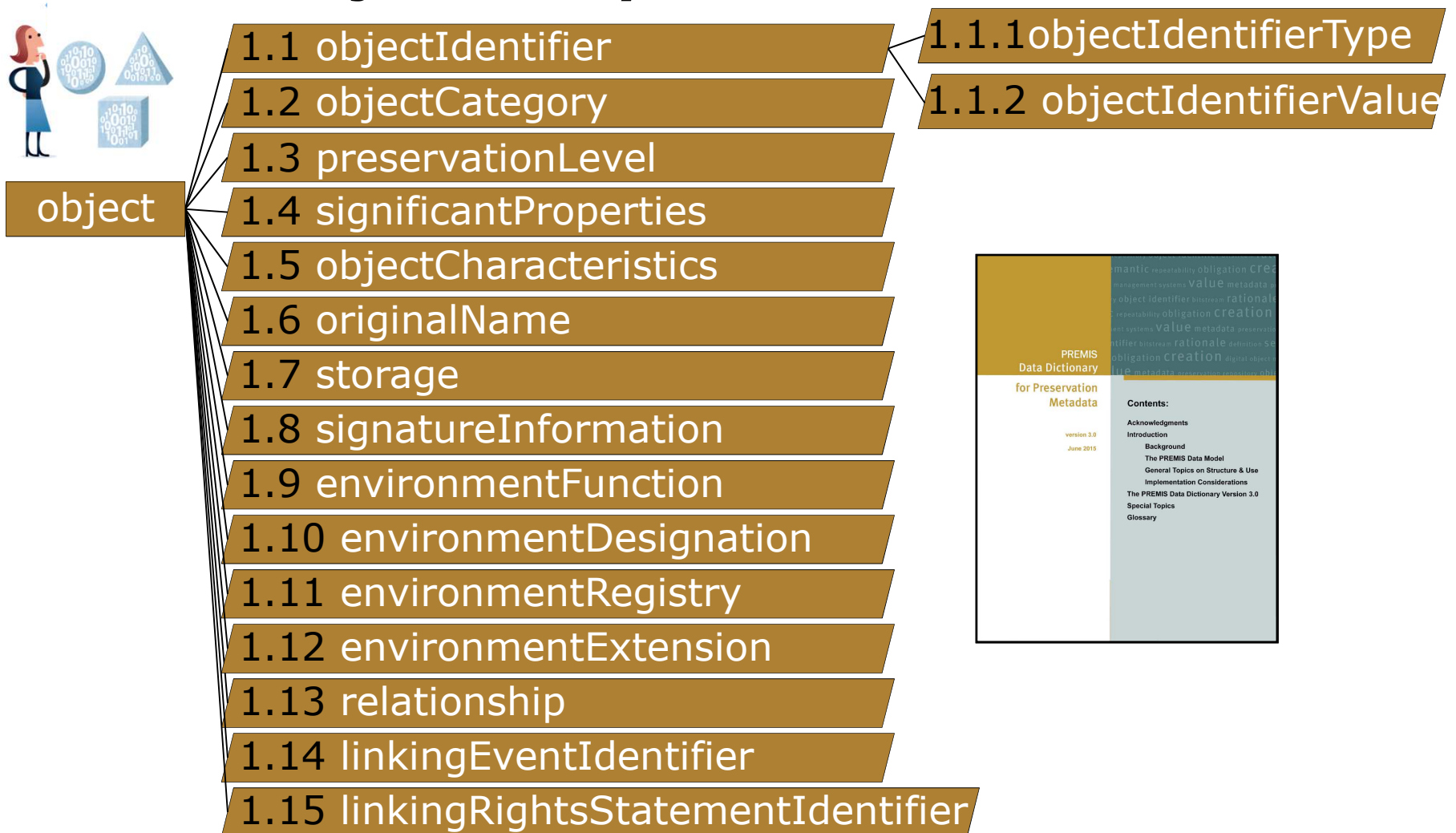
what software or  
hardware should  
be used to handle  
the object?



## PREMIS Object Entity – Semantic Units



### PREMIS Object Entity – Semantic Units



### Sample Data Dictionary Entry



1.1 objectIdentifier

1.1.1 objectIdentifierType

1.1.2 objectIdentifierValue

Object category (type)

Intellectual Entity

Representation

File

Bitstream

Repeatable (R)

Not Repeatable (NR)

Optional (O)

Mandatory (M)

Semantic unit	1.1 objectIdentifier		
Semantic components	1.1.1 objectIdentifierType 1.1.2 objectIdentifierValue		
Definition	A designation used to identify the Object uniquely within the preservation repository system in which it is stored.		
Rationale	Each Object held in the preservation repository must have a unique identifier to allow other entities to refer to it and to relate it to descriptive, technical, and other metadata unambiguously.		
Data constraint	Container		
Object category	Intellectual Entity / Representation	File	Bitstream
Applicability	Applicable	Applicable	Applicable
Repeatability	Repeatable	Repeatable	Repeatable
Obligation	Mandatory	Mandatory	Mandatory
Creation / Maintenance notes	An identifier may be created by the repository system at the time of ingest, or it may be created or assigned outside of the repository and submitted with an object as metadata. Similarly, identifiers can be generated automatically or manually.		
Usage notes	The <i>objectIdentifier</i> is mandatory for all Objects stored. The <i>objectIdentifier</i> is repeatable in order to allow both repository		

## Sample Data Dictionary Entry



1.1 objectIdentifier

1.1.1 objectIdentifierType

1.1.2 objectIdentifierValue

*Object category (type)*

Intellectual Entity

Representation

File

Bitstream

Repeatable (R)

Not Repeatable (NR)

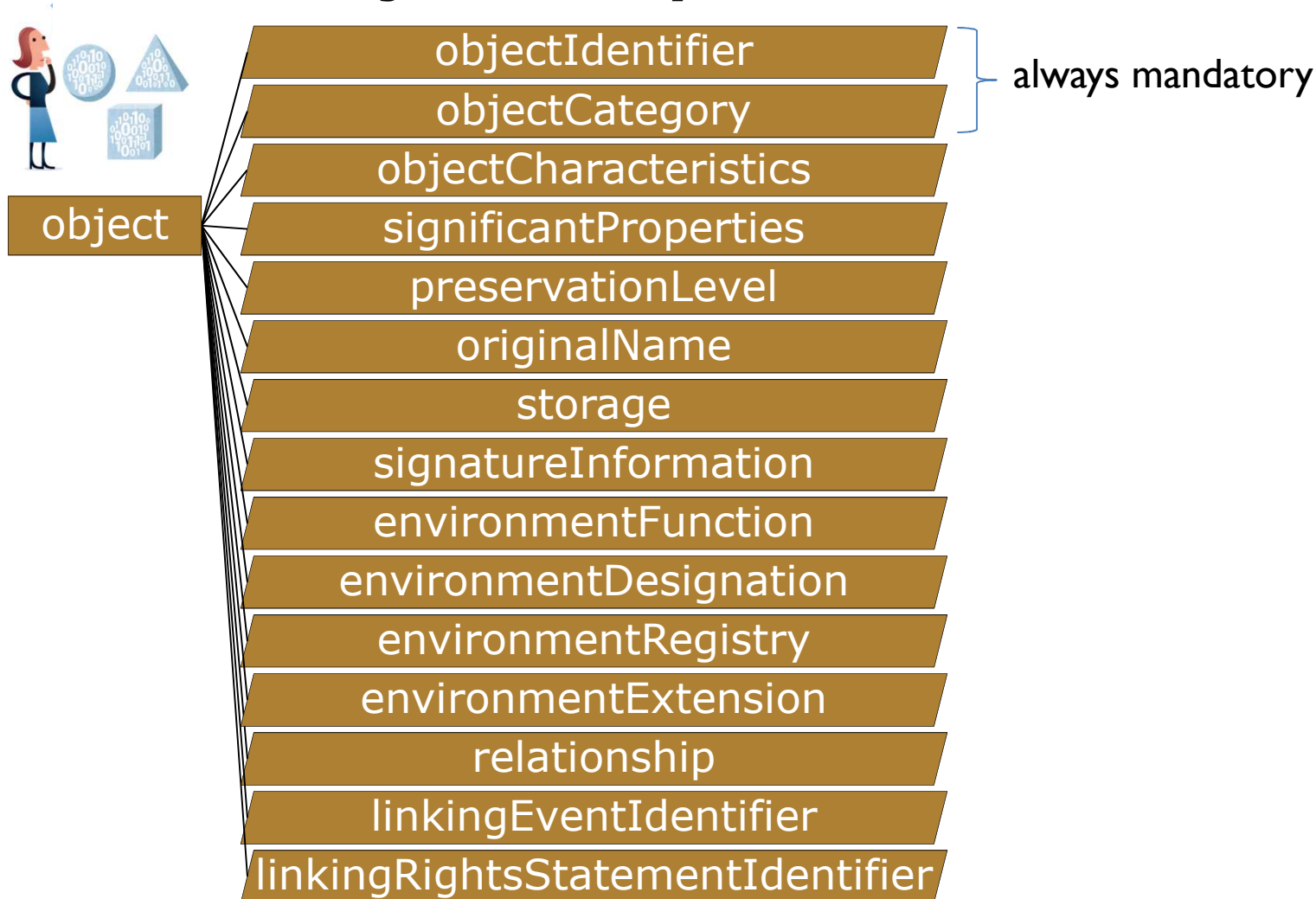
Optional (O)

Mandatory (M)

PREMIS Tutorial at iPres2019

Semantic unit	1.1 objectIdentifier
Semantic components	1.1.1 objectIdentifierType
Entity semantic units	
Definition	<i>NB: Semantic units are applicable for Intellectual Entities, Representations, Files and Bitstreams unless otherwise indicated.</i>
Relationship	<p>1.1 objectIdentifier (M, R)</p> <p>1.1.1 objectIdentifierType (M, NR)</p> <p>1.1.2 objectIdentifierValue (M, NR)</p>
Data	<p>1.2 objectCategory (M, NR)</p> <p>1.3 preservationLevel (O, R) [Intellectual Entity, Representation, File]</p> <p>1.3.1 preservationLevelType (O, NR) [Intellectual Entity, Representation, File]</p> <p>1.3.2 preservationLevelValue (M, NR) [Intellectual Entity, Representation, File]</p> <p>1.3.3 preservationLevelRole (O, NR) [Intellectual Entity, Representation, File]</p> <p>1.3.4 preservationLevelRationale (O, R) [Intellectual Entity, Representation, File]</p> <p>1.3.5 preservationLevelDateAssigned (O, NR) [Intellectual Entity, Representation, File]</p>
Object	
Application	
Representation	<p>1.4 significantProperties (O, R)</p> <p>1.4.1 significantPropertiesType (O, NR)</p> <p>1.4.2 significantPropertiesValue (O, NR)</p> <p>1.4.3 significantPropertiesExtension (O, R)</p>
Obligation	
Creation	<p>1.5 objectCharacteristics (M, R) [File, Bitstream]</p> <p>1.5.1 compositionLevel (O, NR) [File, Bitstream]</p> <p>1.5.2 fixity (O, R) [File, Bitstream]</p> <p>1.5.2.1 messageDigestAlgorithm (M, NR) [File, Bitstream]</p> <p>1.5.2.2 messageDigest (M, NR) [File, Bitstream]</p> <p>1.5.2.3 messageDigestOriginator (O, NR) [File, Bitstream]</p> <p>1.5.3 size (O, NR) [File, Bitstream]</p>
Maintenance	
Usage	

## PREMIS Object Entity – Semantic Units



## objectCategory



- Values:
  - intellectual entity
  - representation
  - file
  - bitstream
- Implemented as an xsi:type in PREMIS XML-schema so not explicitly recorded

**<premis>**

**<object** xsi:type="file">

...  
**</object>**

...  
**</premis>**

## objectCategory (types of objects)



**INTELLECTUAL ENTITY:** a distinct intellectual or artistic creation that is considered relevant to a designated community in the context of digital preservation.



**REPRESENTATION:** set of files, including structural metadata, that, taken together, constitute a complete rendering of an Intellectual Entity.

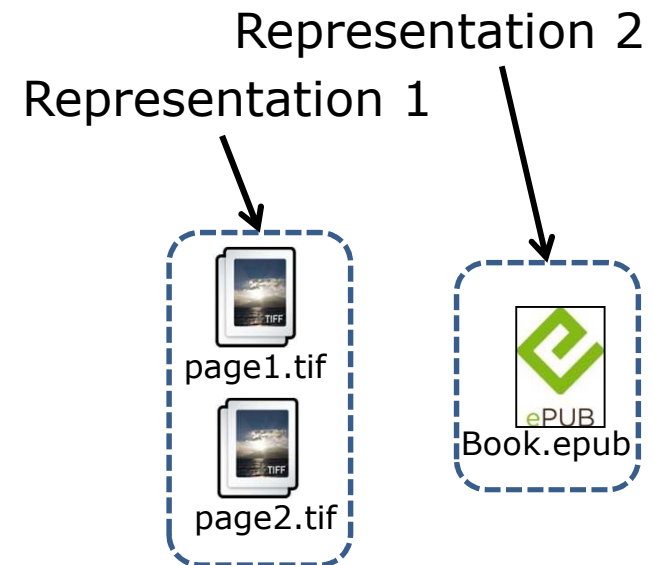


**FILE:** named and ordered sequence of bytes that is known by an operating system (file-streams (files within files) are considered files since they can be rendered alone)

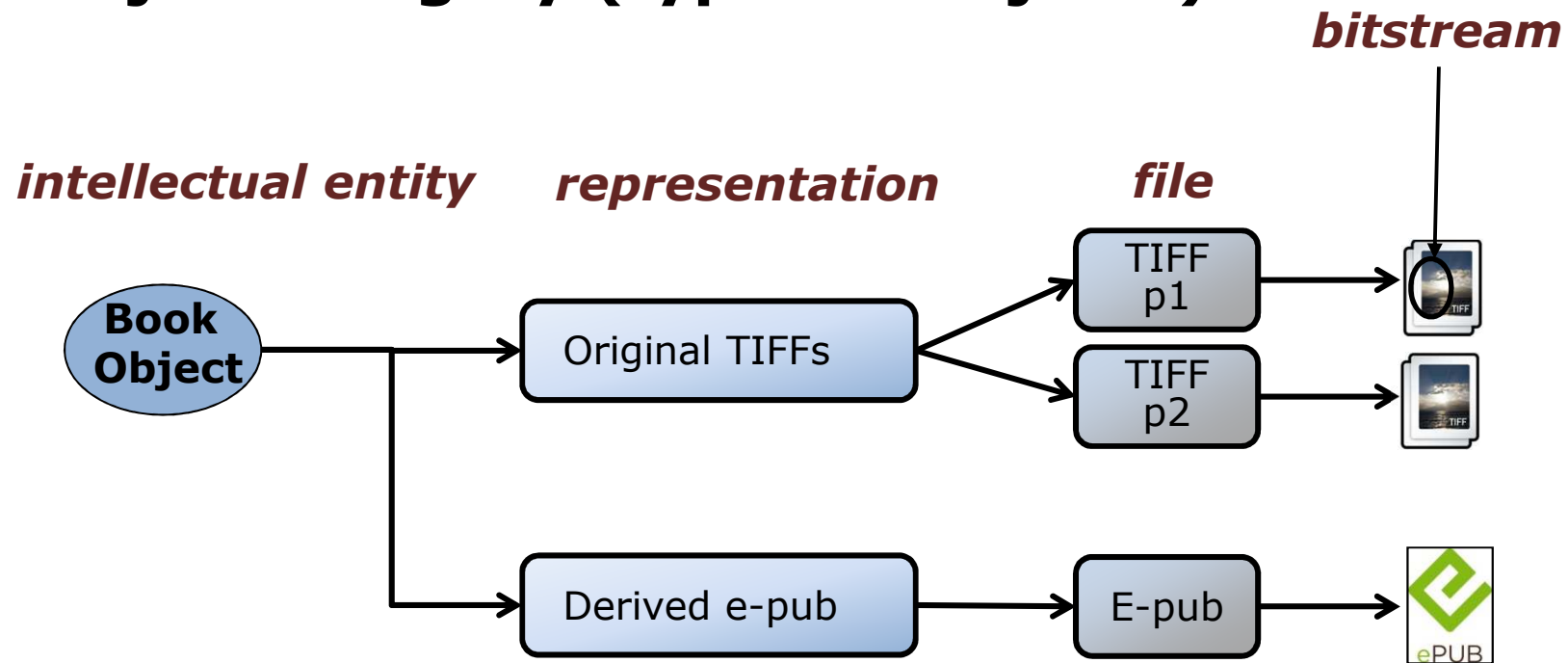


**BITSTREAM:** data within a file with properties relevant for preservation purposes (but needs additional structure or reformatting to be stand-alone file)

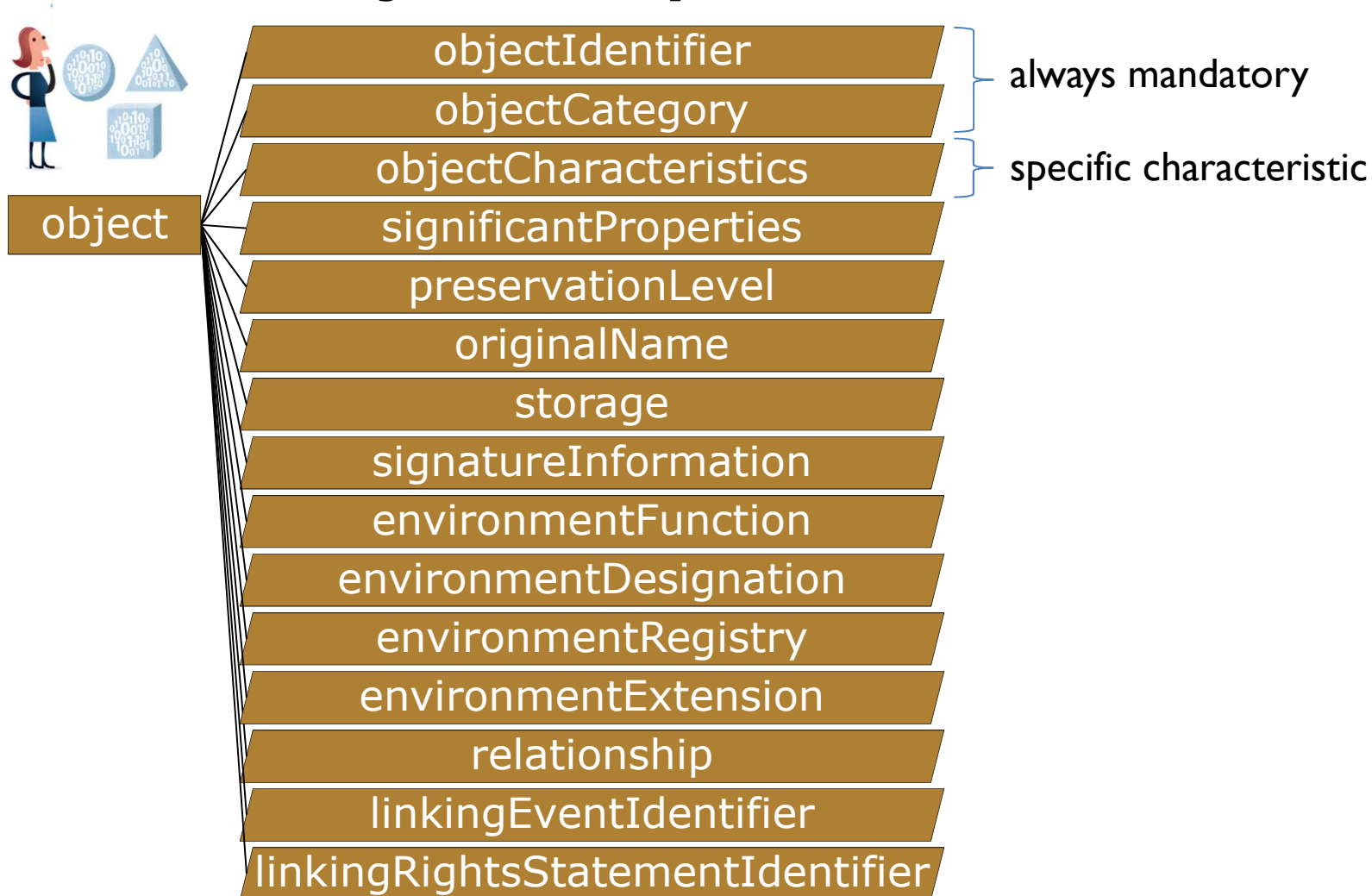
← A literary “work”



## objectCategory (types of objects)



## PREMIS Object Entity



## objectCharacteristics

[mandatory for file or bitstream]

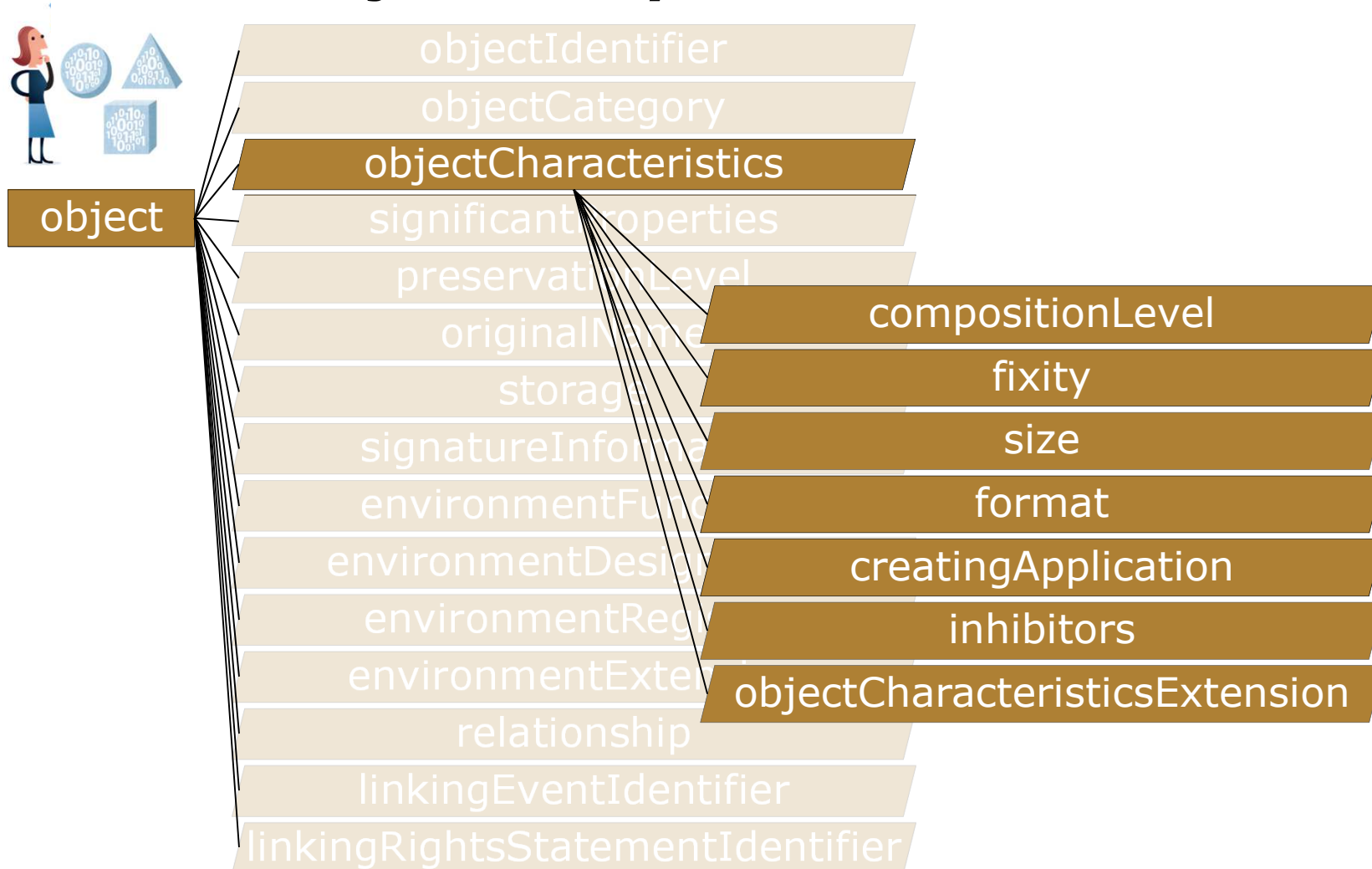
- Technical properties common to all/most file formats, not format specific

### Looking at files

FILE = a named sequence of bytes

- chapter1.pdf
- photo.tiff
- mapofBerlin.jp2
- Can be zero or more bytes
- Has a file format
- Has access permissions and file system statistics such as size and modification date

## PREMIS Object Entity



## objectCharacteristicsExtension



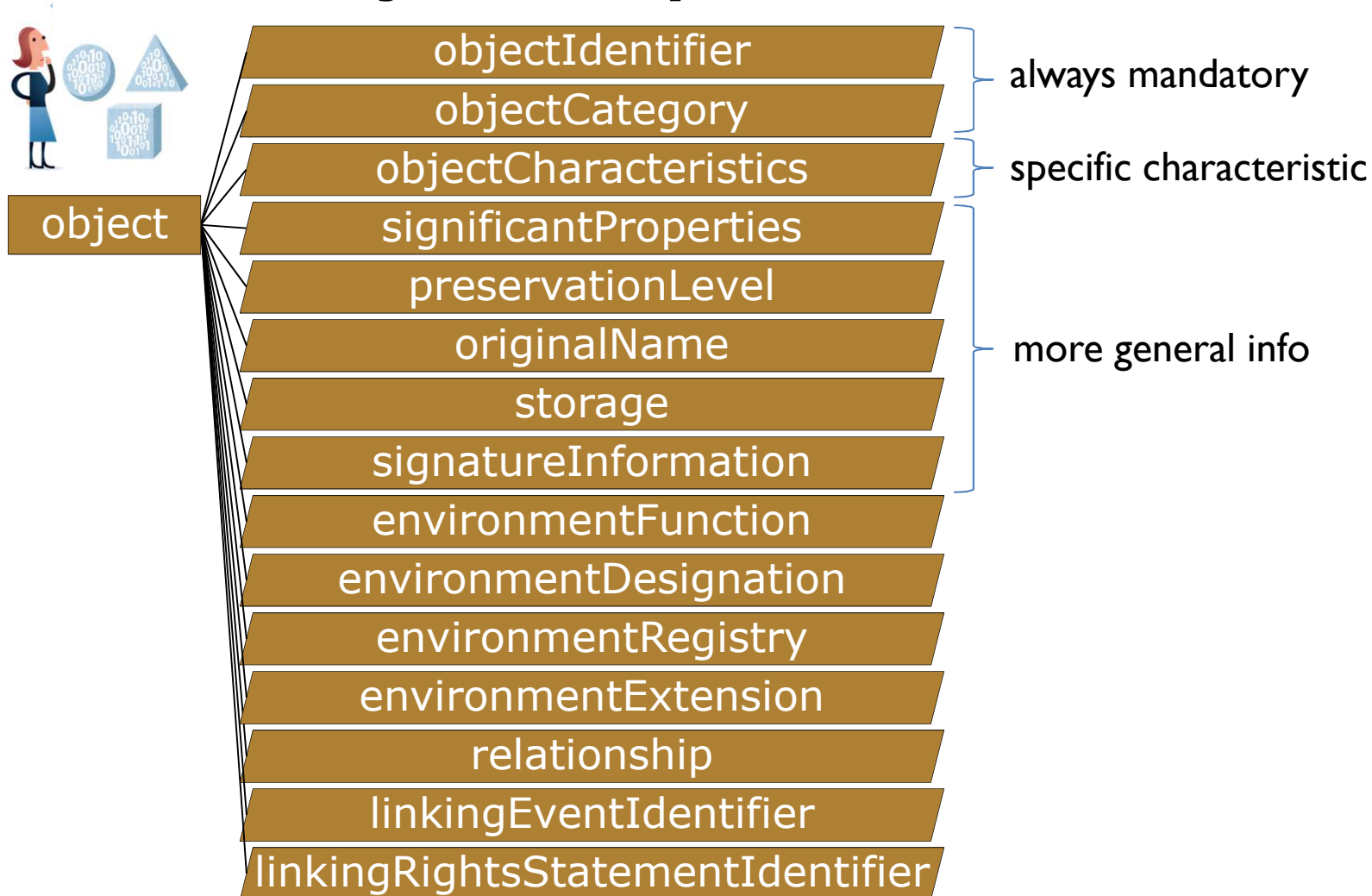
Container to include externally defined semantic units –  
e.g. for more granularity

Might contain format specific metadata for a file – e.g.  
technical metadata for still images (MIX)

## objectCharacteristicsExtension - example

```
<premis> ...  
  <object xsi:type="file">  
    <objectCharacteristics> ...  
      <objectCharacteristicsExtension>  
        <mix:mix xsi:schemaLocation=  
          "... http://www.loc.gov/standards/mix/mix20/mix20.xsd">  
          ...  
          <mix:BasicImageInformation>  
            <mix:BasicImageCharacteristics>  
              <mix:imageWidth>5894</mix:imageWidth>  
              <mix:imageHeight>7768</mix:imageHeight>  
              ...  
            </mix:BasicImageCharacteristics>  
          </mix:BasicImageInformation>  
          ...  
          <mix:mix>  
            </objectCharacteristicsExtension> ...  
          <objectCharacteristics> ...  
        </object> ...  
      </premis>
```

## PREMIS Object Entity – Semantic Units



## preservationLevel

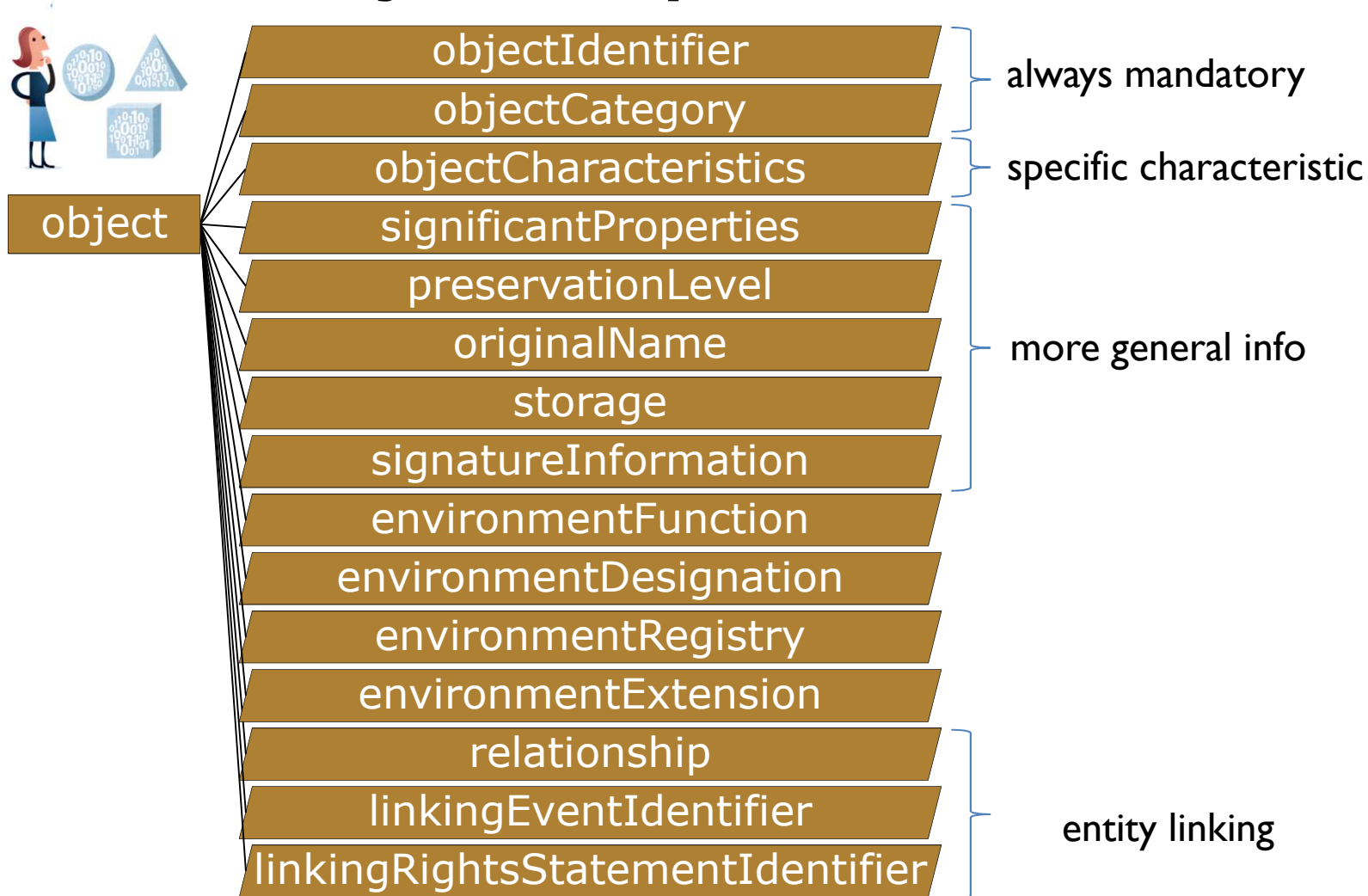
**What preservation treatment/strategy the repository plans for this object**

- Varying preservation options dependent on factors such as value, uniqueness, preservability of format
- A business rule only relevant in a given repository

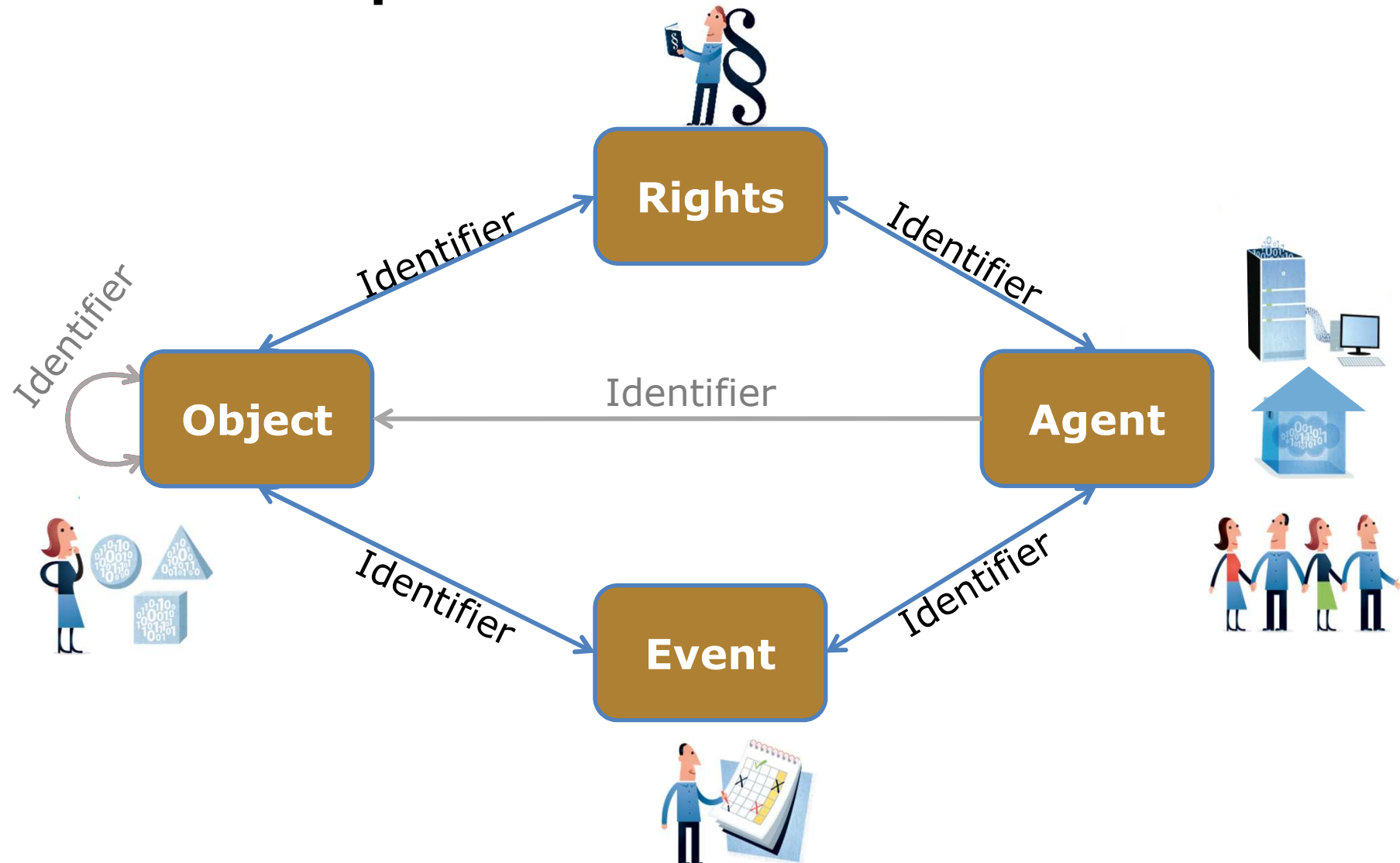
### Contains

- **preservationLevelType**, e.g. **logicalStrategy** or **BitSafety**
- **preservationLevelValue**, e.g. **migration** or **High**
- **preservationLevelRole** (context), e.g. **intention** or **requirement**
- **preservationLevelRationale**, when differs from policy
- **preservationLevelDateAssigned** when Level was assigned

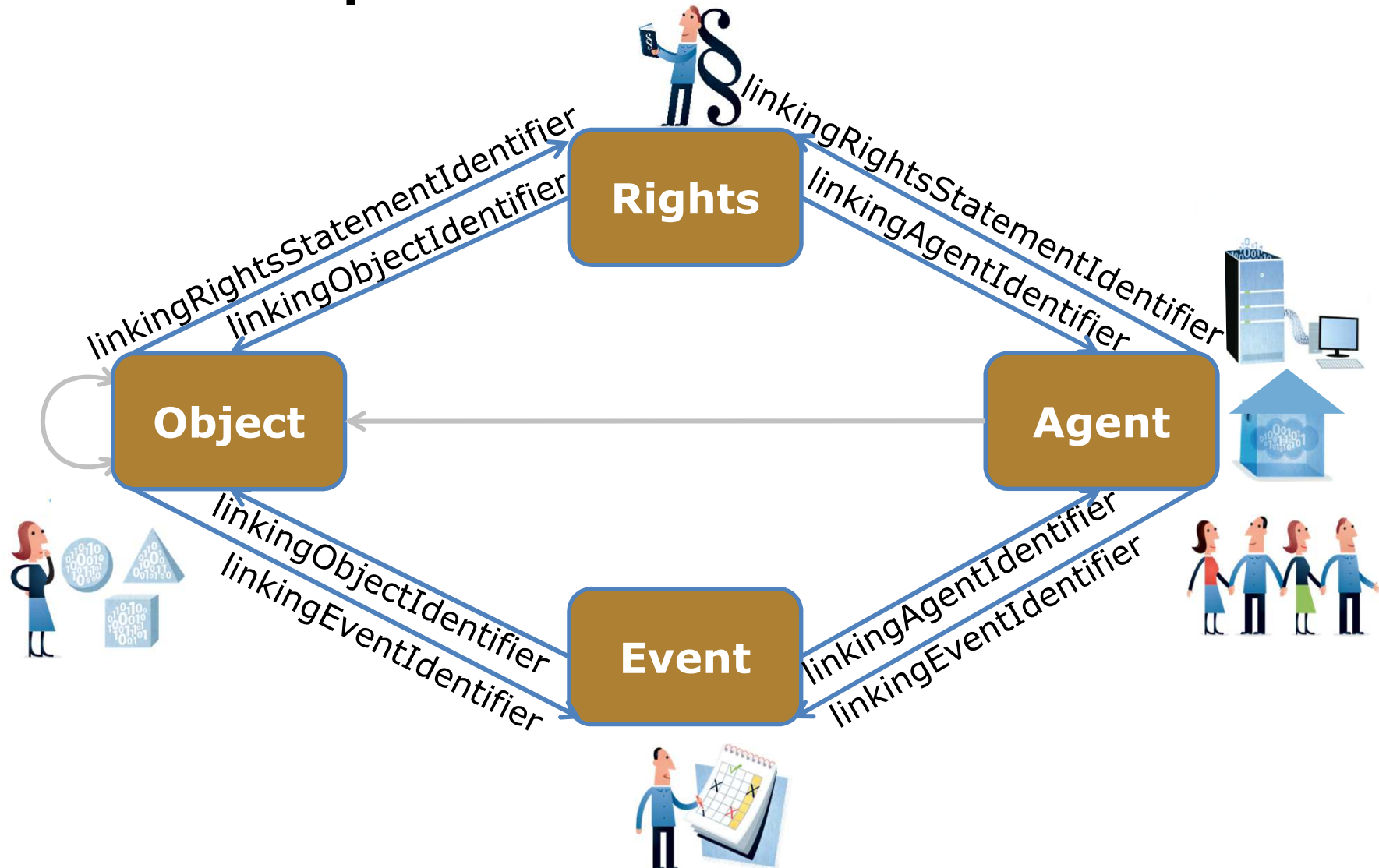
## PREMIS Object Entity – Semantic Units



## Relationships: Semantic Unit Identifiers



## Relationships: Semantic Unit Identifiers



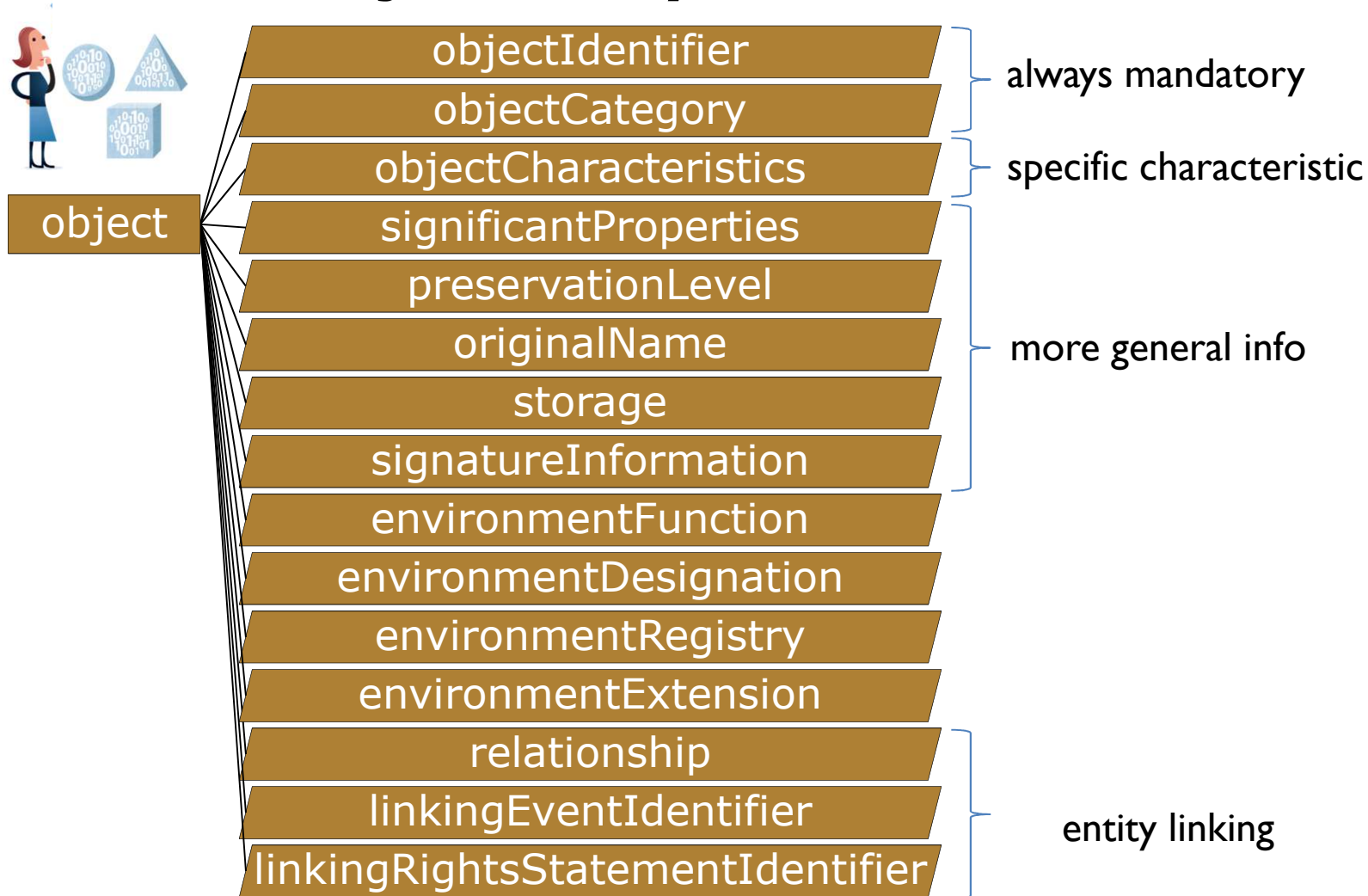
## Linking Objects with Agents and Events

- **linkingRightsStatementIdentifier**
  - **linkingRightsStatementIdentifierType**
  - **linkingRightsStatementIdentifierValue**
- **linkingEventIdentifier**
  - **linkingEventIdentifierType**
  - **linkingEventIdentifierValue**

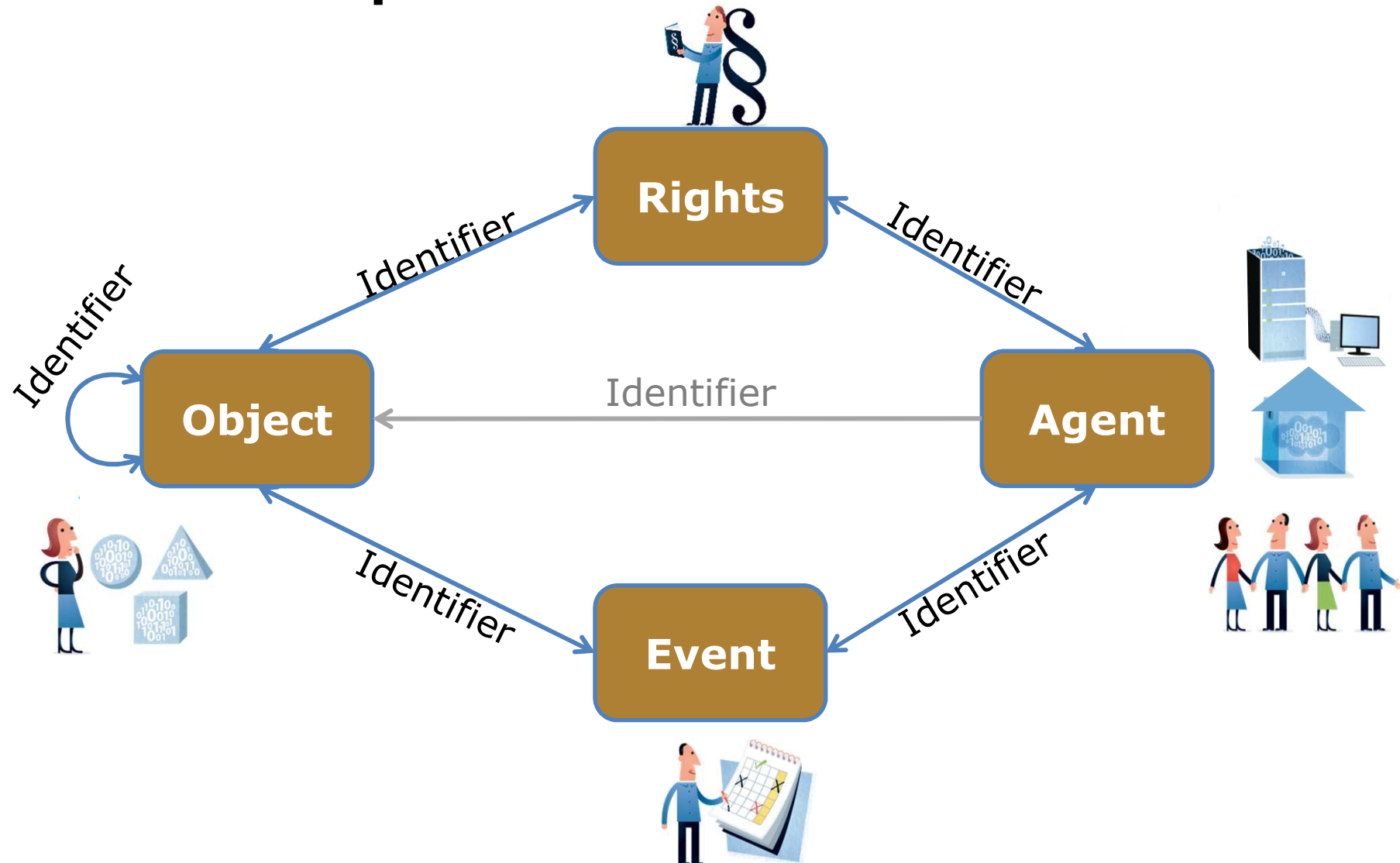
**Likewise for the other entities**

**except** there may be possibility of specifying roles, e.g. for Event because the same Agent may have a different Role in the digital Archive system

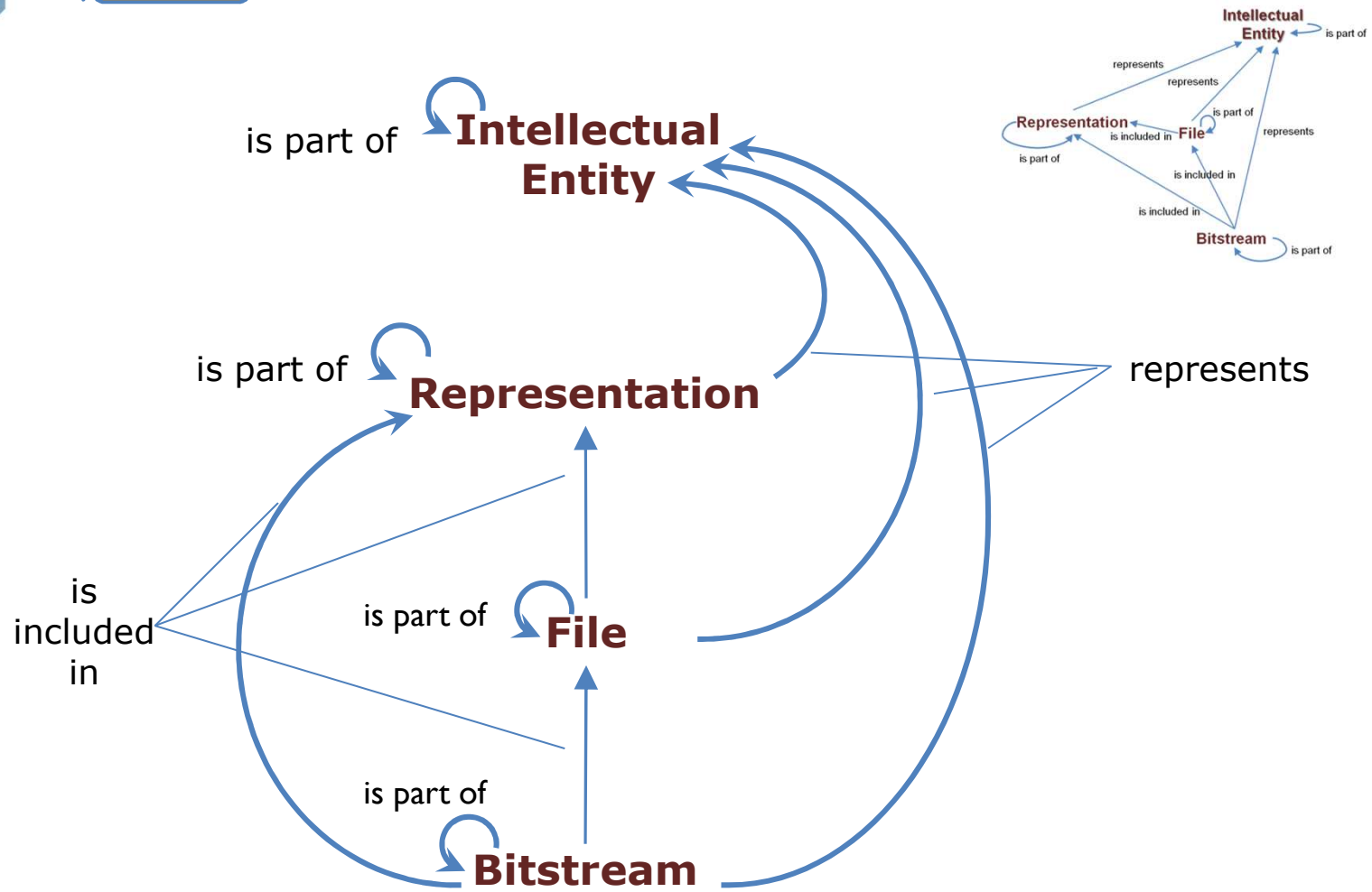
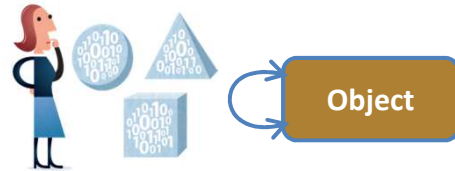
## PREMIS Object Entity – Semantic Units



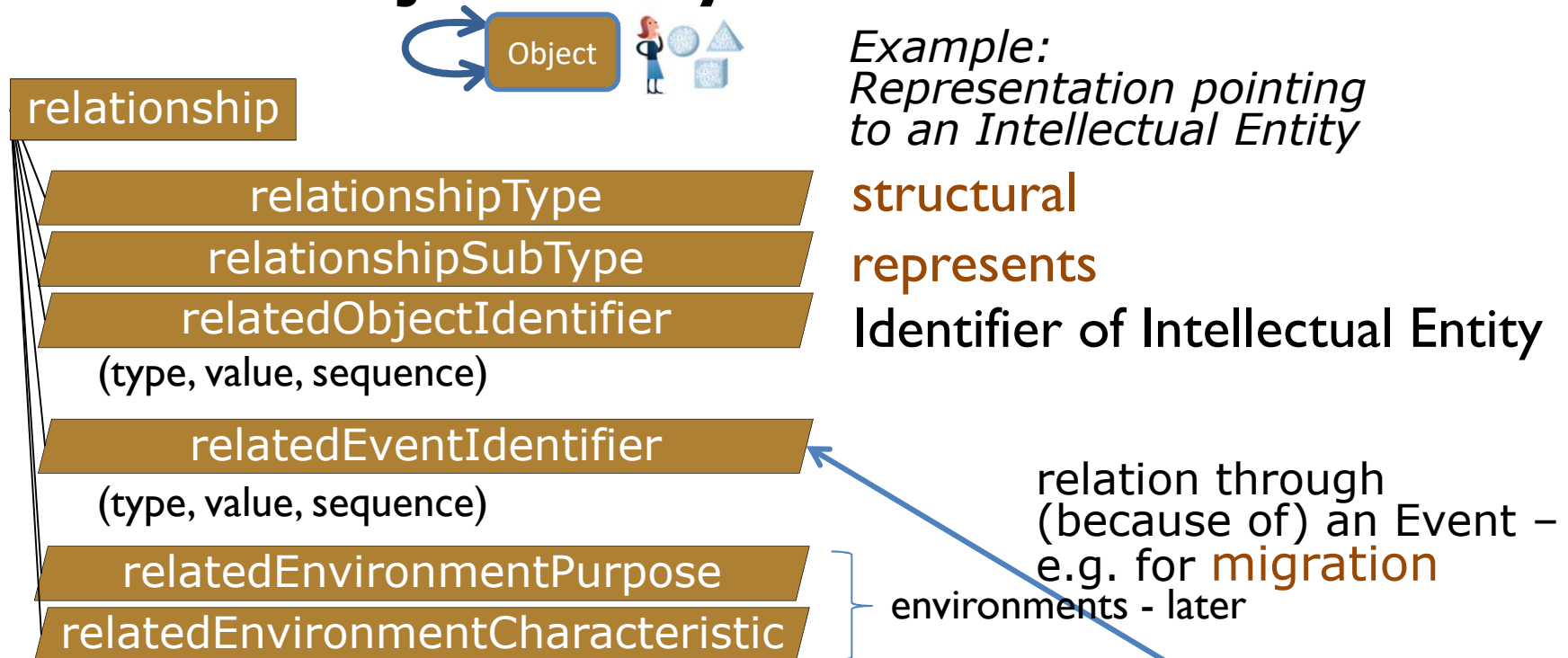
## Relationships: Semantic Unit Identifiers



### Objects and their interrelations



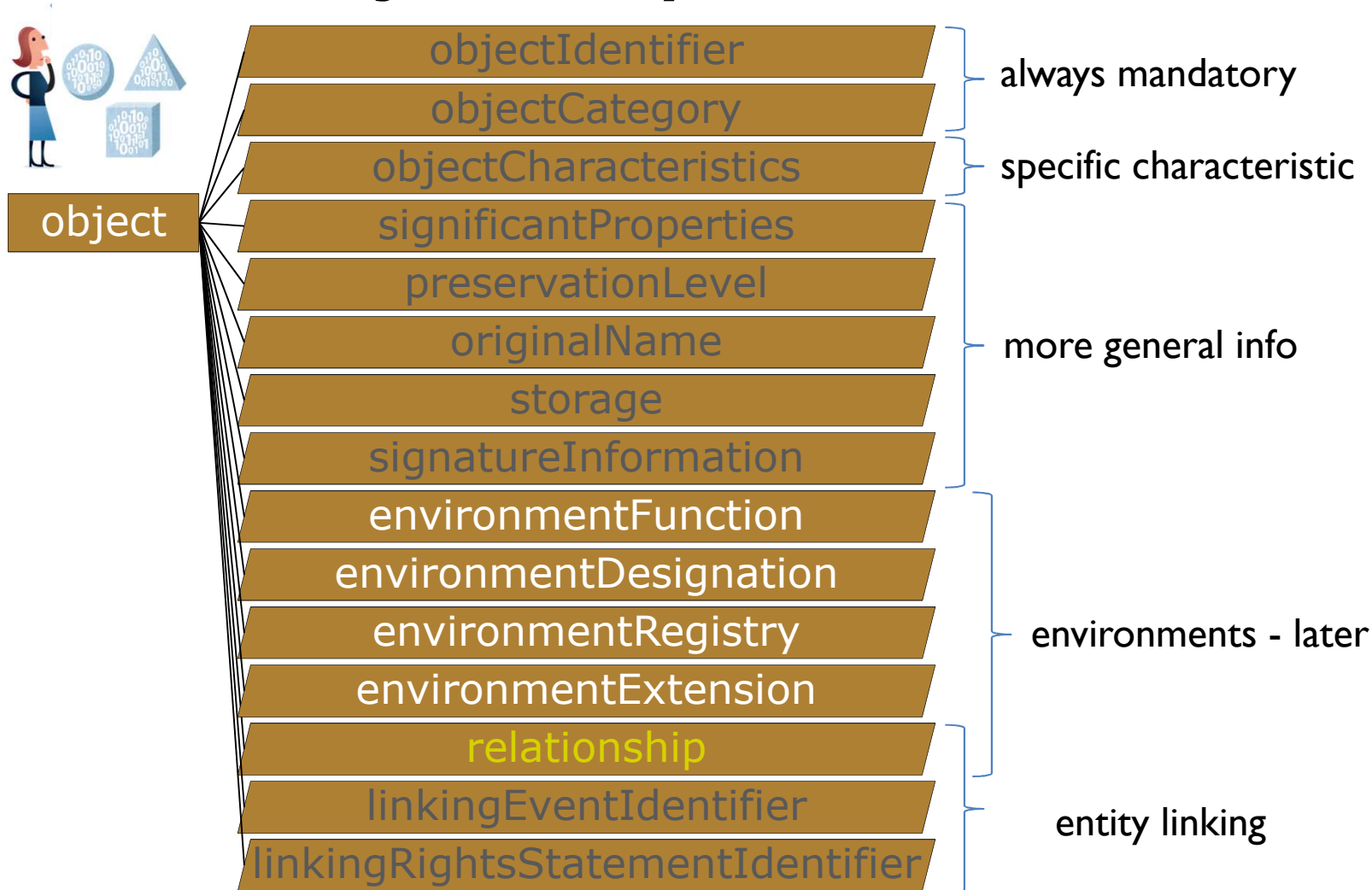
## PREMIS Object Entity – Semantic Units



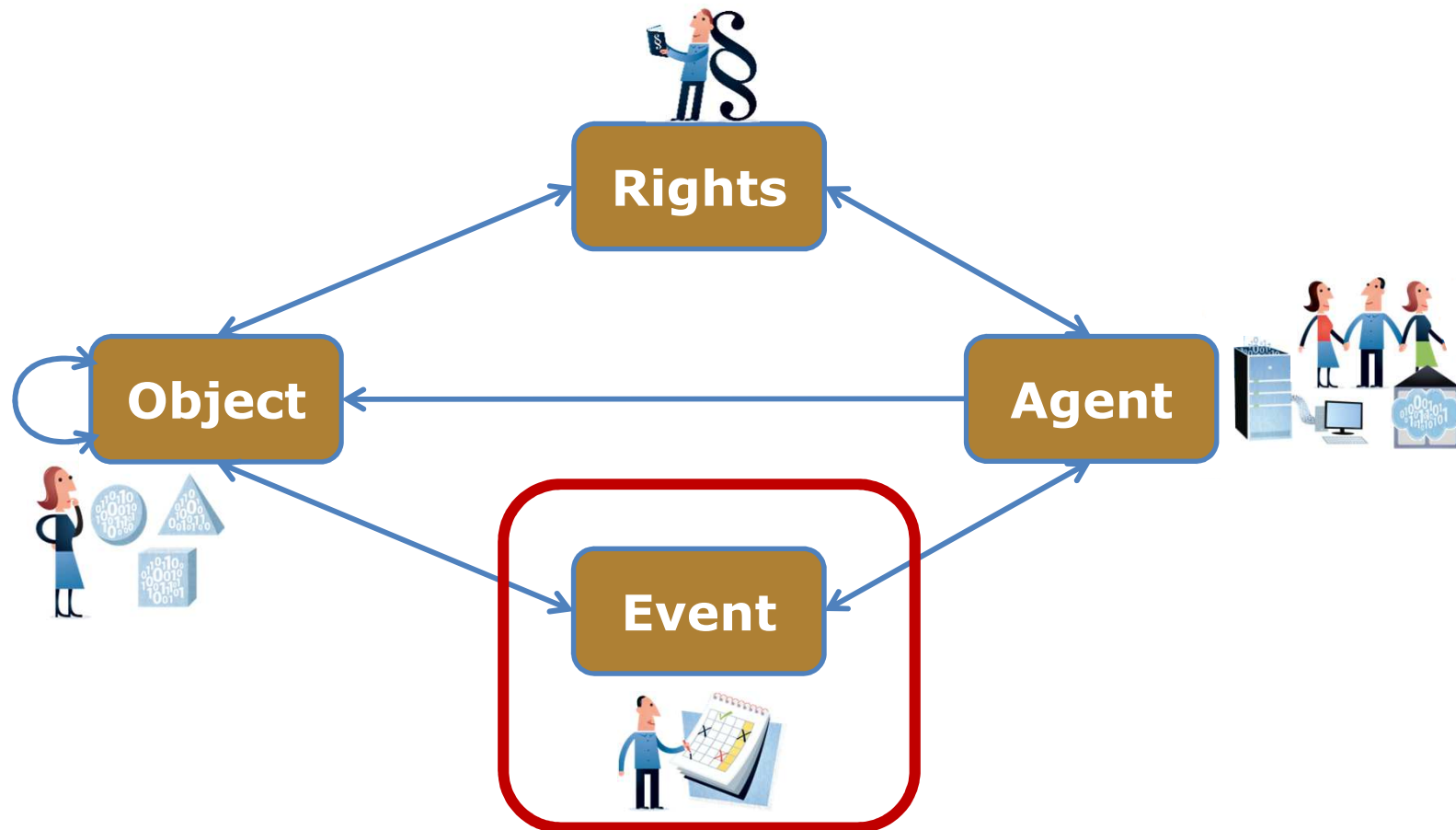
Objects can be associated with Events in two ways:

- If the Object has an associated Event with **relationship**
- If the Object has an associated Event with **no relationship** to a second Object, e.g. **ingest**: use **linkingEventIdentifier**

### PREMIS Object Entity – Semantic Units

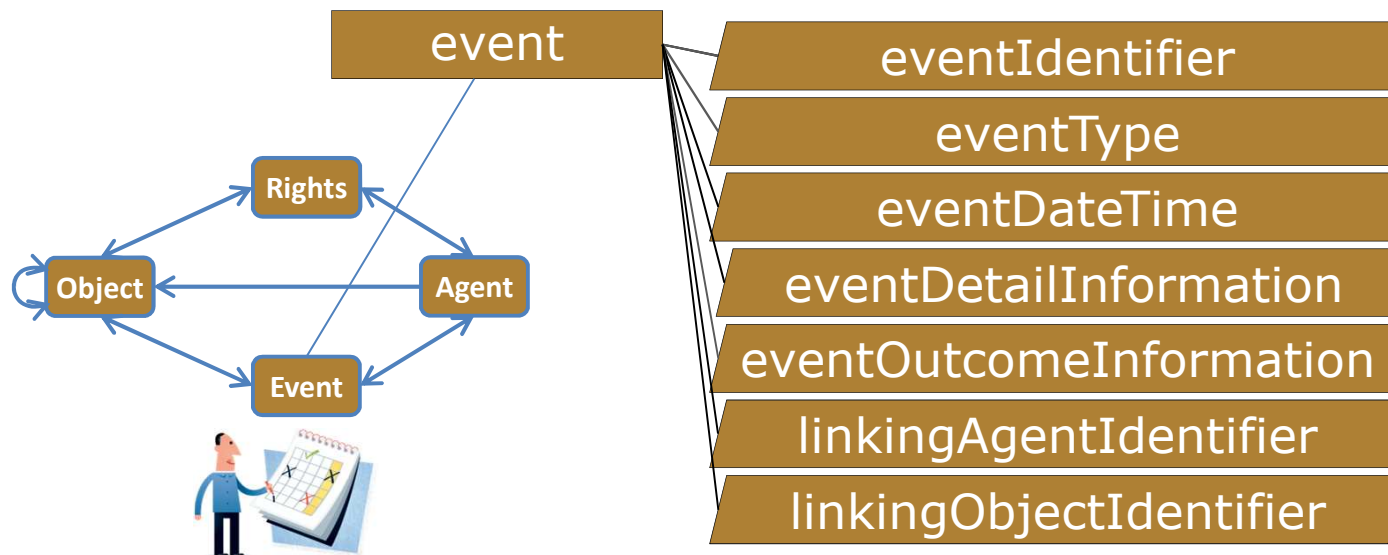


## Properties of Entities - Semantic units



## PREMIS Event Entity

- Mandatory semantic units are: *eventIdentifier*, *eventType*, and *eventDateTime*.
- Must be related to one or more Objects.
- Can be related to one or more Agents.



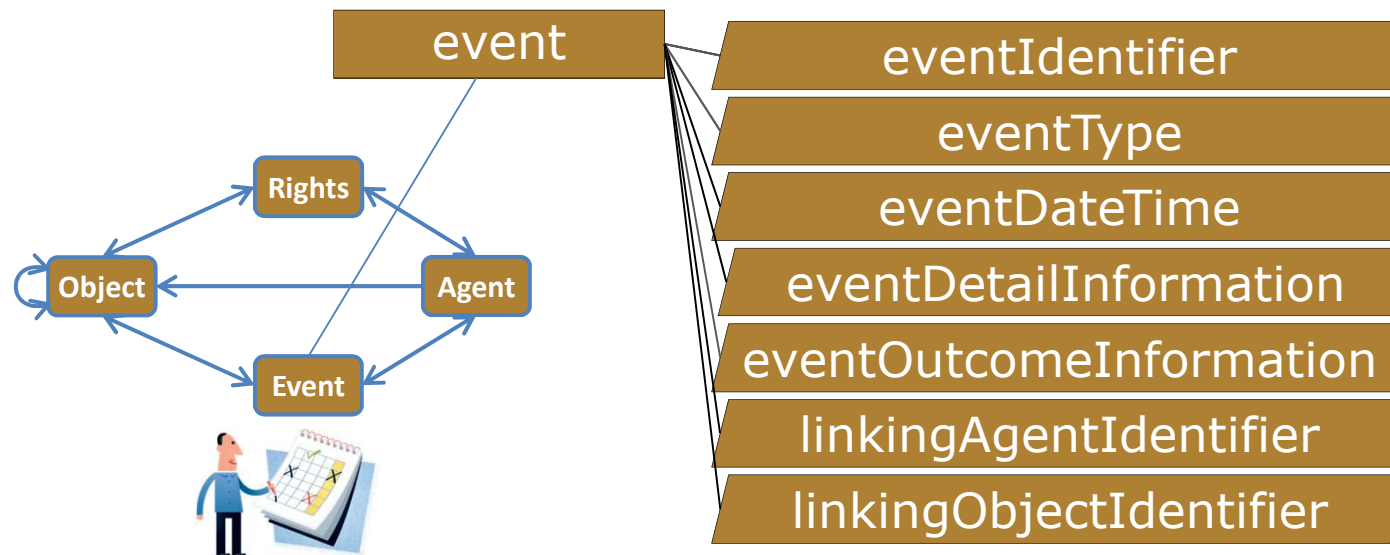
## eventType

- Names the event

**Ingestion**      **Validation**      **Virus check**  
**Message digest calculation**      **Compression**  
**migration**      **Fixity check**      **Decompression**      ...

- Recommended to use a controlled vocabulary, e.g.  
<http://id.loc.gov/vocabulary/preservation/eventType.html>
- Could use coded values
- Granularity is implementation-specific

## PREMIS Event Entity



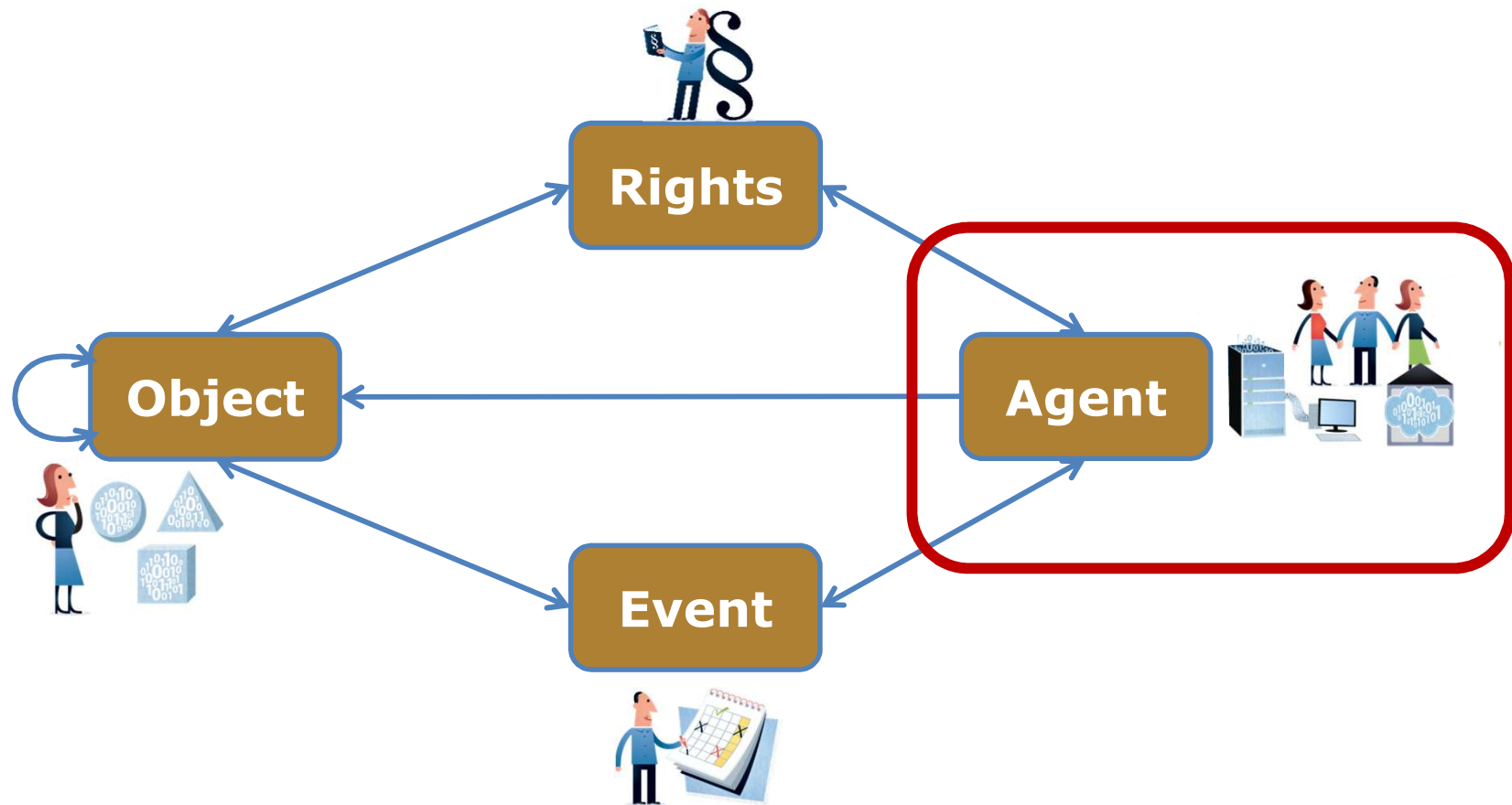
## eventOutcomeInformation

Structures outcome information

Contains

- **eventOutcome**  
*intended to be codes, e.g. 00 meaning OK*
- **eventOutcomeDetail**  
*more granular information in*
  - **eventOutcomeDetailNote**  
*textual form e.g. "new file successfully created"*
  - **eventOutcomeDetailExtension**  
*information using other schemas*

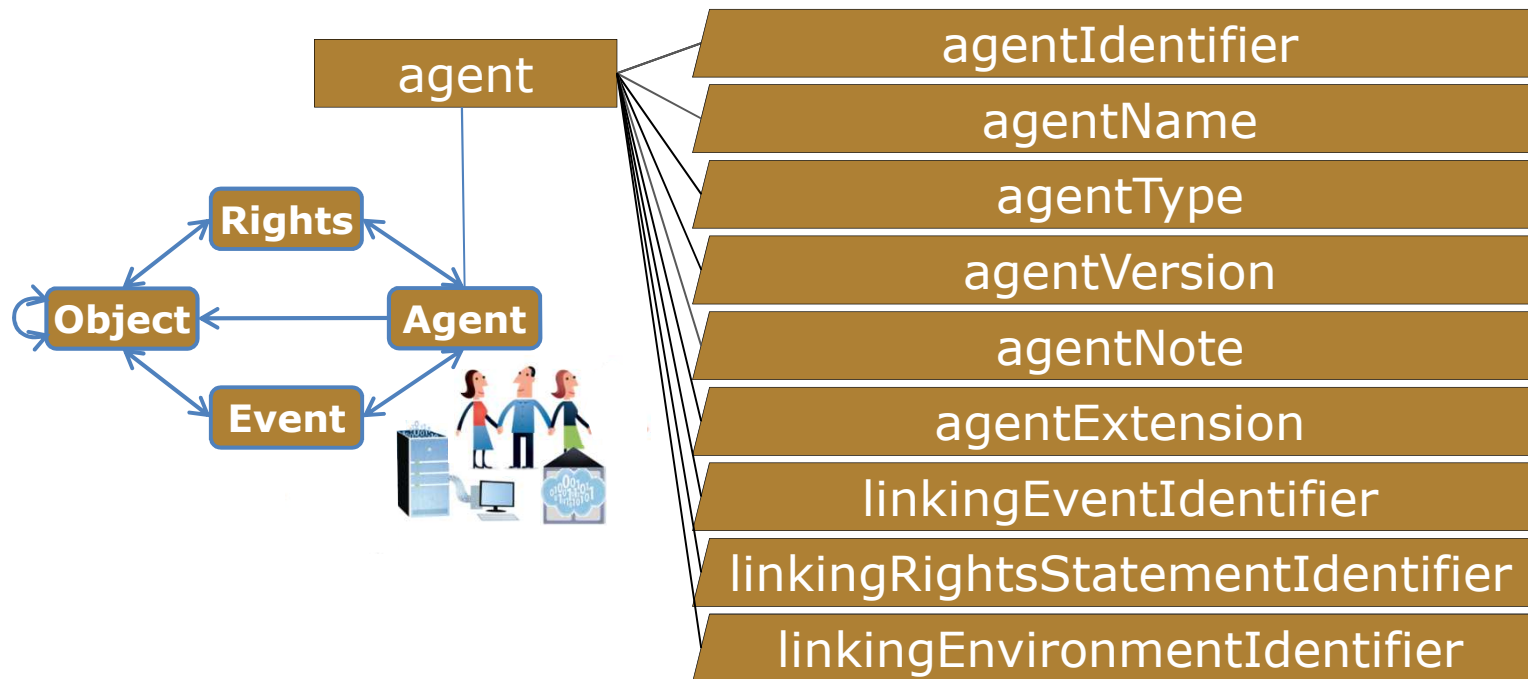
## Properties of Entities - Semantic units



## PREMIS Agent Entity

The only mandatory semantic unit is *agentIdentifier*

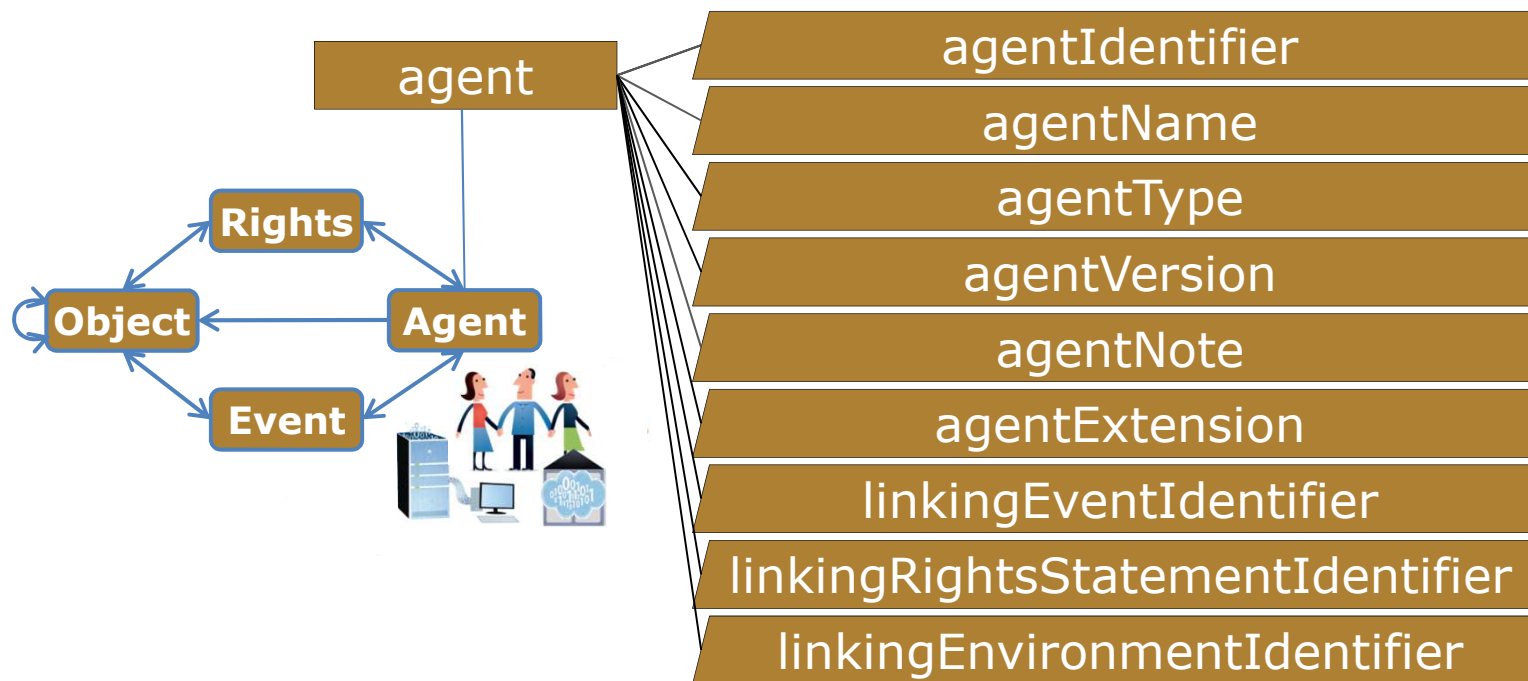
- May hold or grant one or more Rights.
- May carry out, authorize, or compel one or more Events.
- May create or act upon one or more Objects through an Event or with respect to a Rights statement.



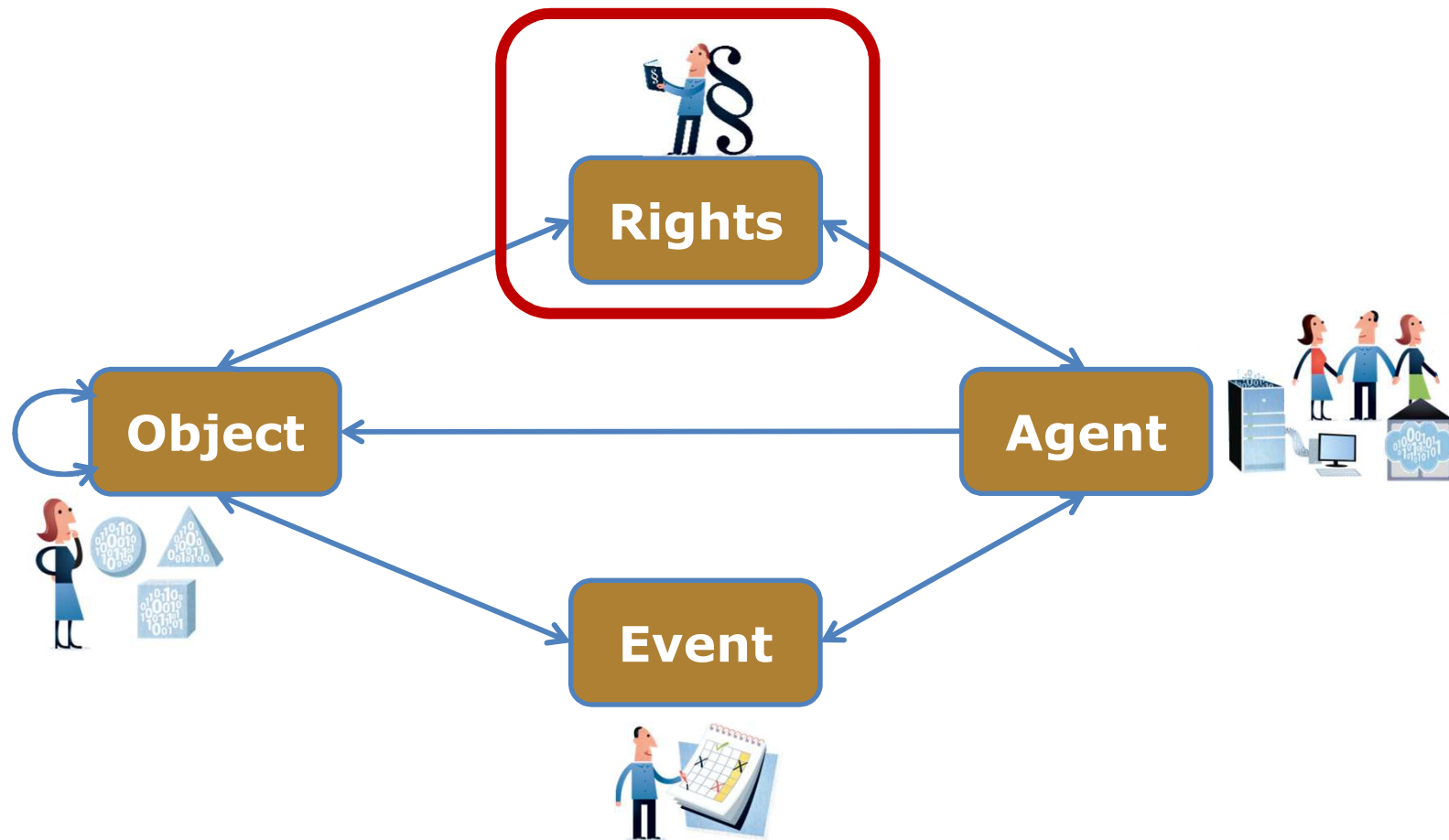
## agentType

- Can use controlled vocabulary, e.g.  
<http://id.loc.gov/vocabulary/preservation/agentType.html>
  - hardware
  - organization
  - person
  - software

## PREMIS Agent Entity



## Properties of Entities - Semantic units



### PREMIS Rights Entity

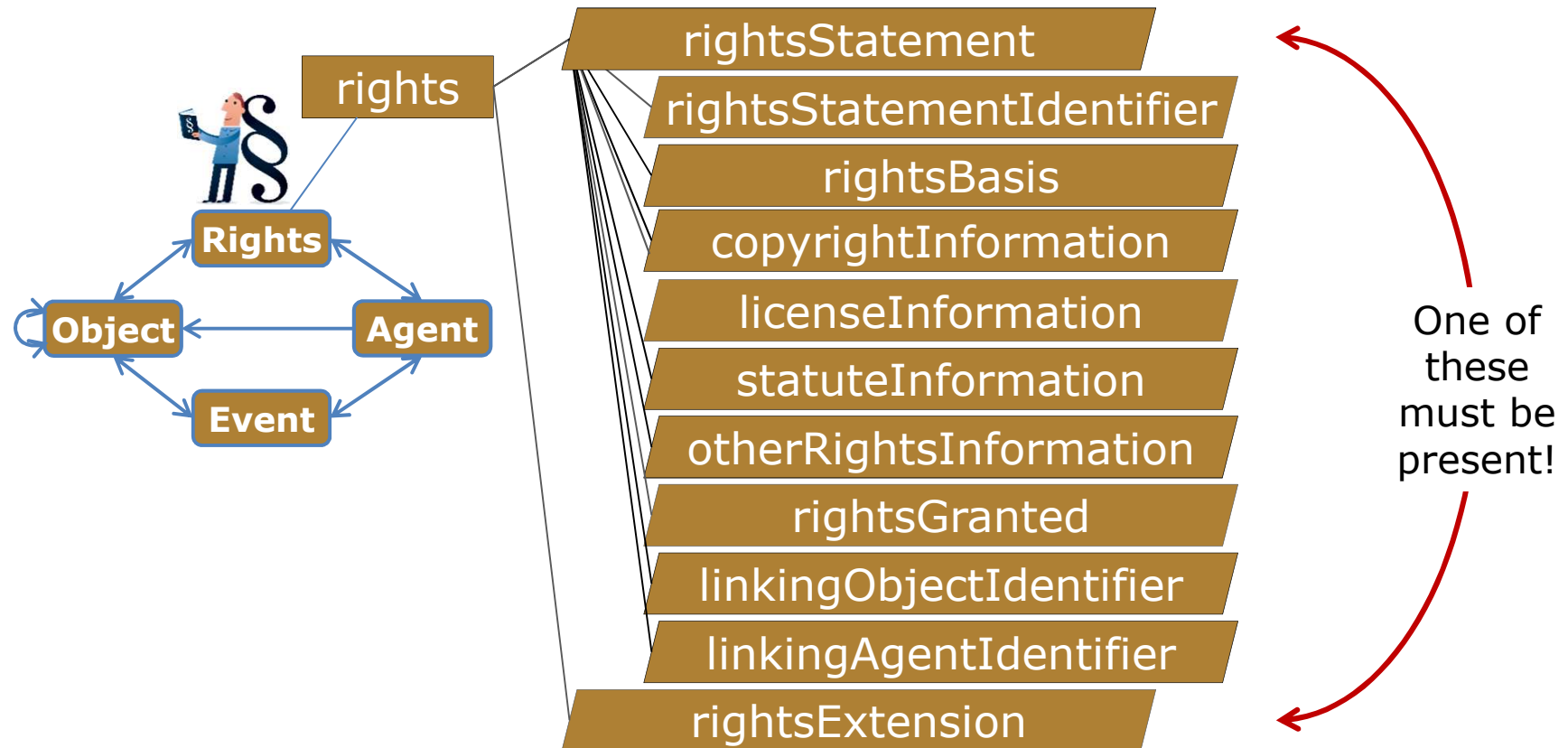
copyright



license



other



## Dependent units about rights



Specifying different types of rights

rightsBasis

copyrightInformation

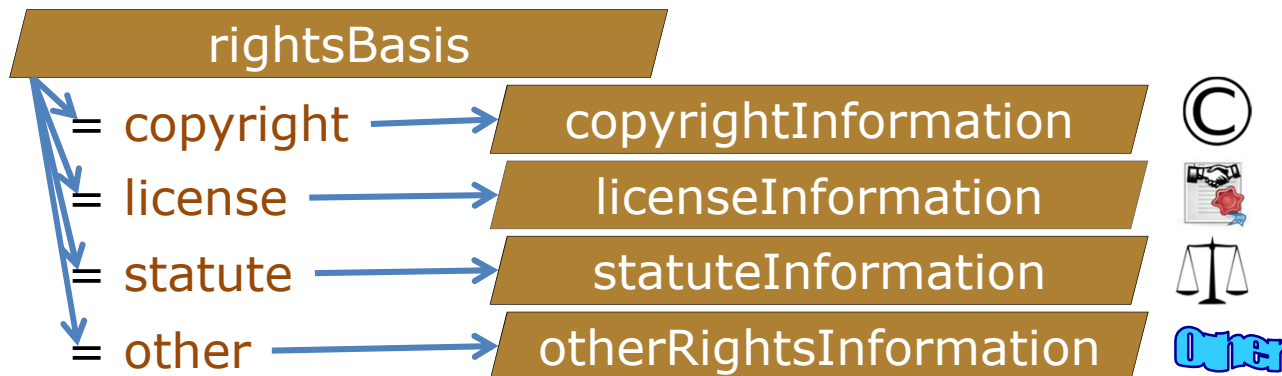
licenseInformation

statuteInformation

otherRightsInformation

## Dependent units about rights

Specifying different types of rights



If more than one basis applies, the entire rights entity should be *repeated*.

## Example rightsBasis and copyrightInformation

**rightsBasis** = **copyright**

### **copyrightInformation**

**copyrightStatus** = **copyrighted**

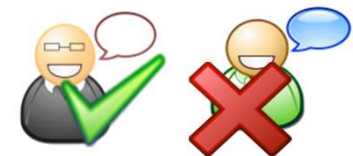
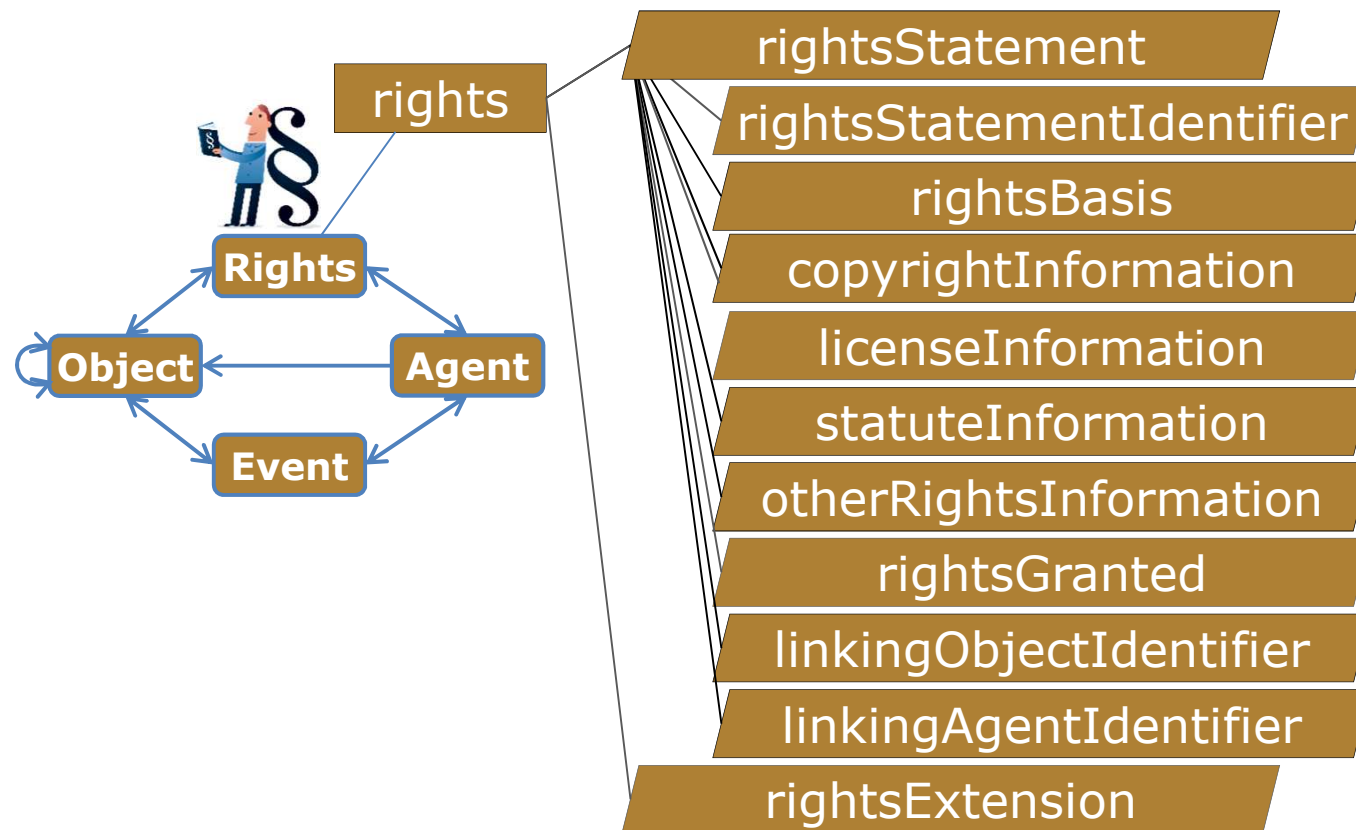
**copyrightJurisdiction** = **us**

**copyrightStatusDeterminationDate** = **2008-09-10**

**copyrightNote** = **Copyright expiration expected in 2022**

**copyrightDocumentationIdentifier** = **[link]**

## PREMIS Rights Entity



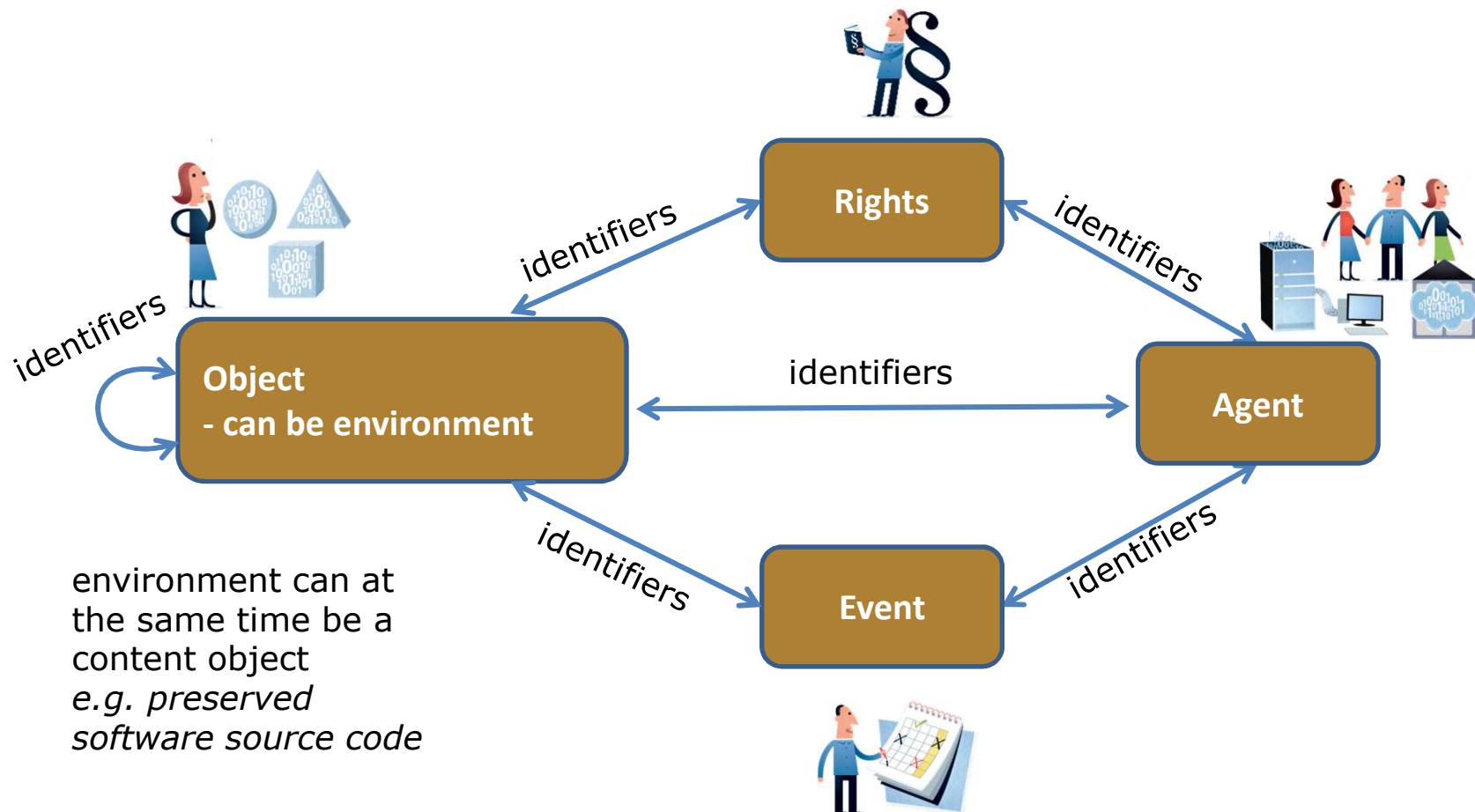
## rightsGranted

- What action is allowed?
- Under what conditions?
- Are there time constraints?

Contains

- **Act** (e.g. *migrate*, *modify*, ... could use *eventType values*)
- **Restriction** (description of condition or limitation on act)
- **termOfGrant** (*start and end dates of rights granted*)
  - **startDate** (e.g. *2005-01-01*)
  - **endDate** (e.g. *2005-01-01*)
- **termOfRestriction** (*start and end dates of restriction granted*)
  - **startDate** (e.g. *2005-01-01*)
  - **endDate** (e.g. *OPEN*)
- **rightsGrantedNote** (*additional inf. about the rights granted*)

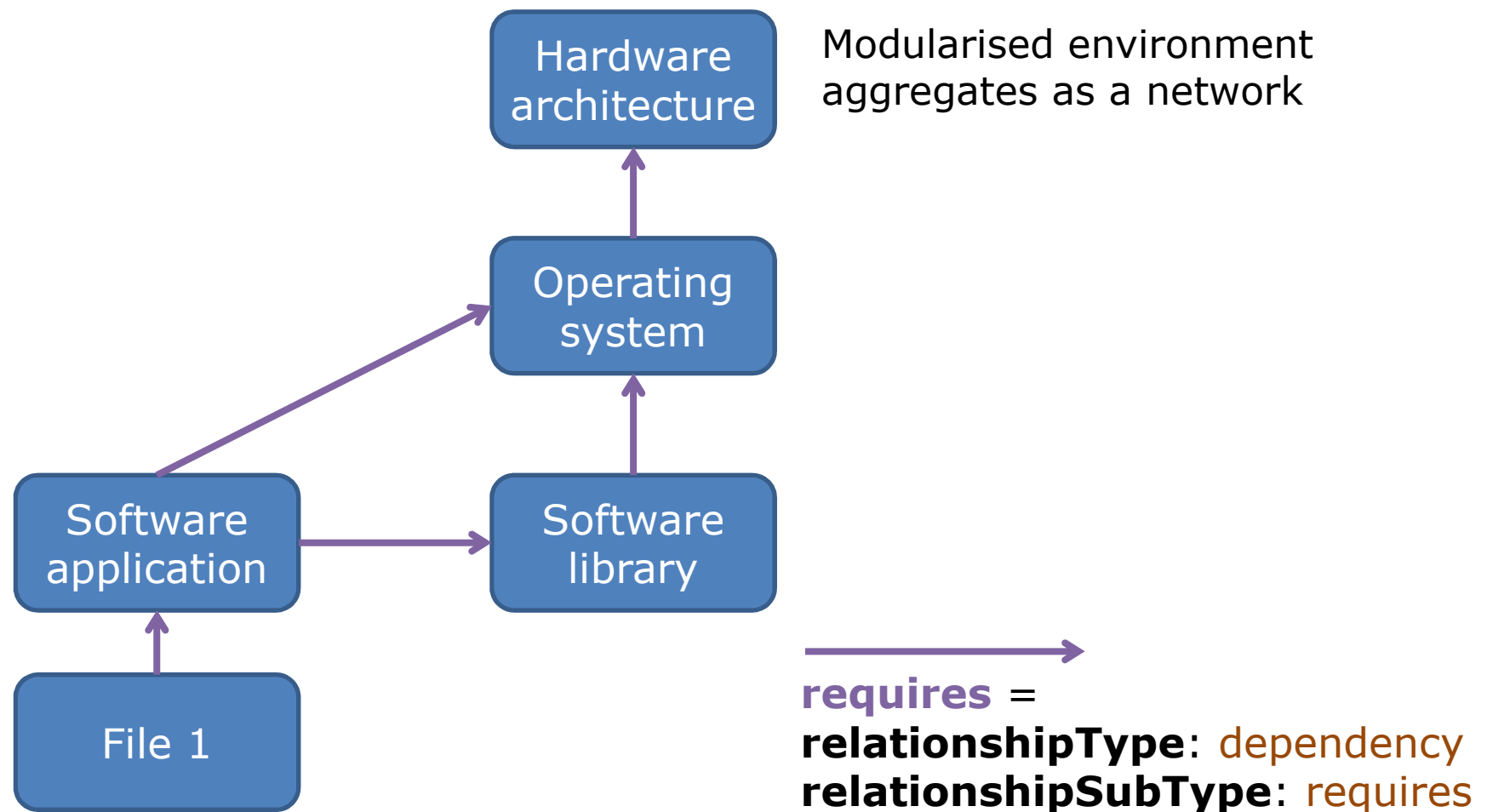
## Data Model - environments



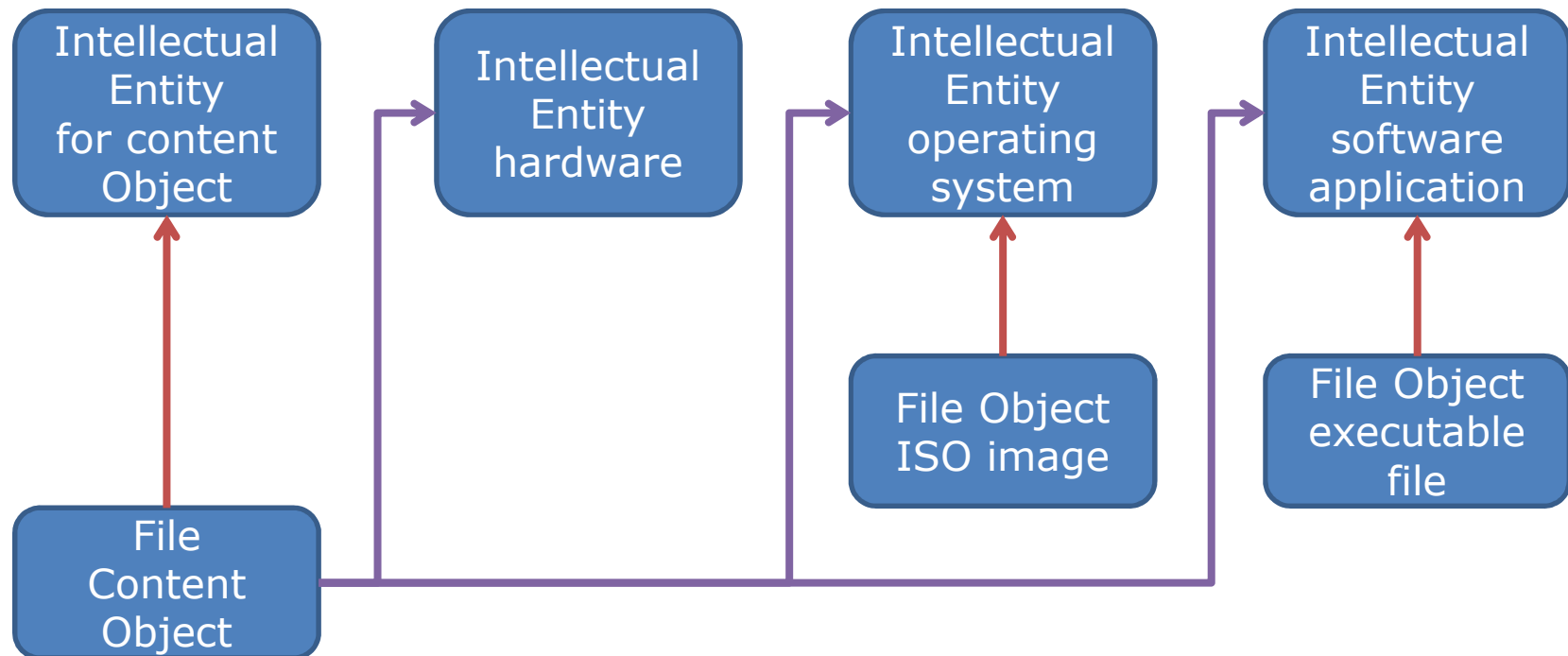
## Environments as independent objects

- What is needed to render or use a content object
  - Operating system
  - Application software
  - Hardware
  - Computing resources

## Example: Environment stack and dependency relationships



**Example:**  
**An object and its rendering environment**



→  
**represents** =  
**relationshipType:** structural  
**relationshipSubType:** represents

→  
**requires** =  
**relationshipType:** dependency  
**relationshipSubType:** requires

agent



agentIdentifier

agentName

agentType

agentVersion

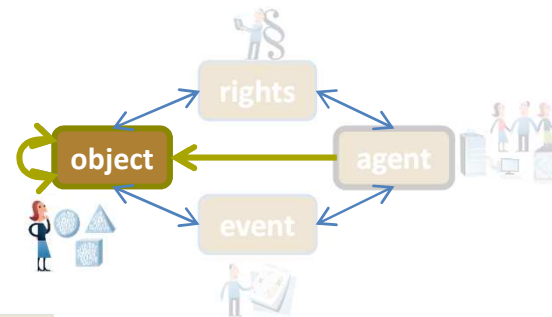
agentNote

agentExtension

linkingEventIdentifier

linkingRightsStatementIdentifier

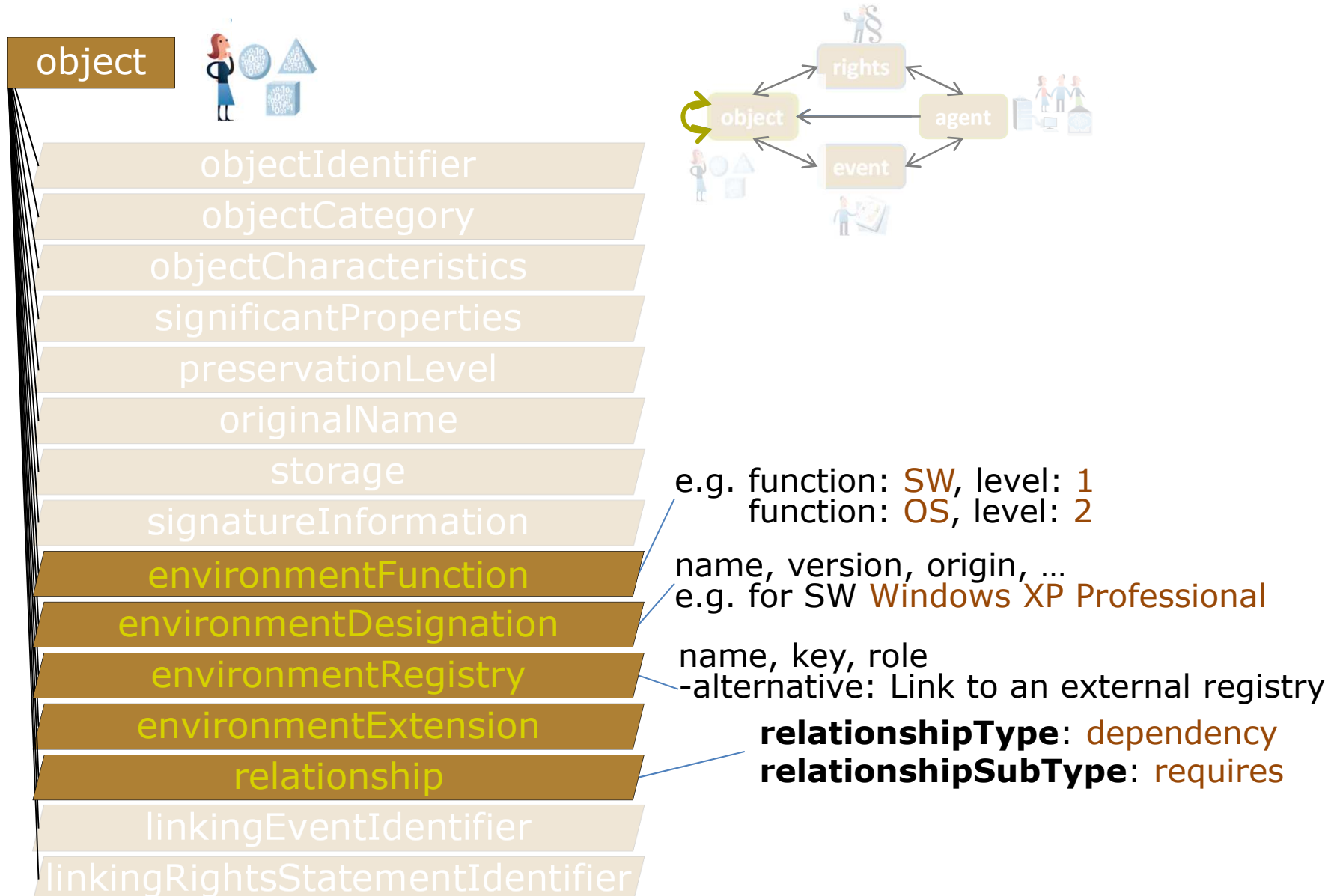
linkingEnvironmentIdentifier

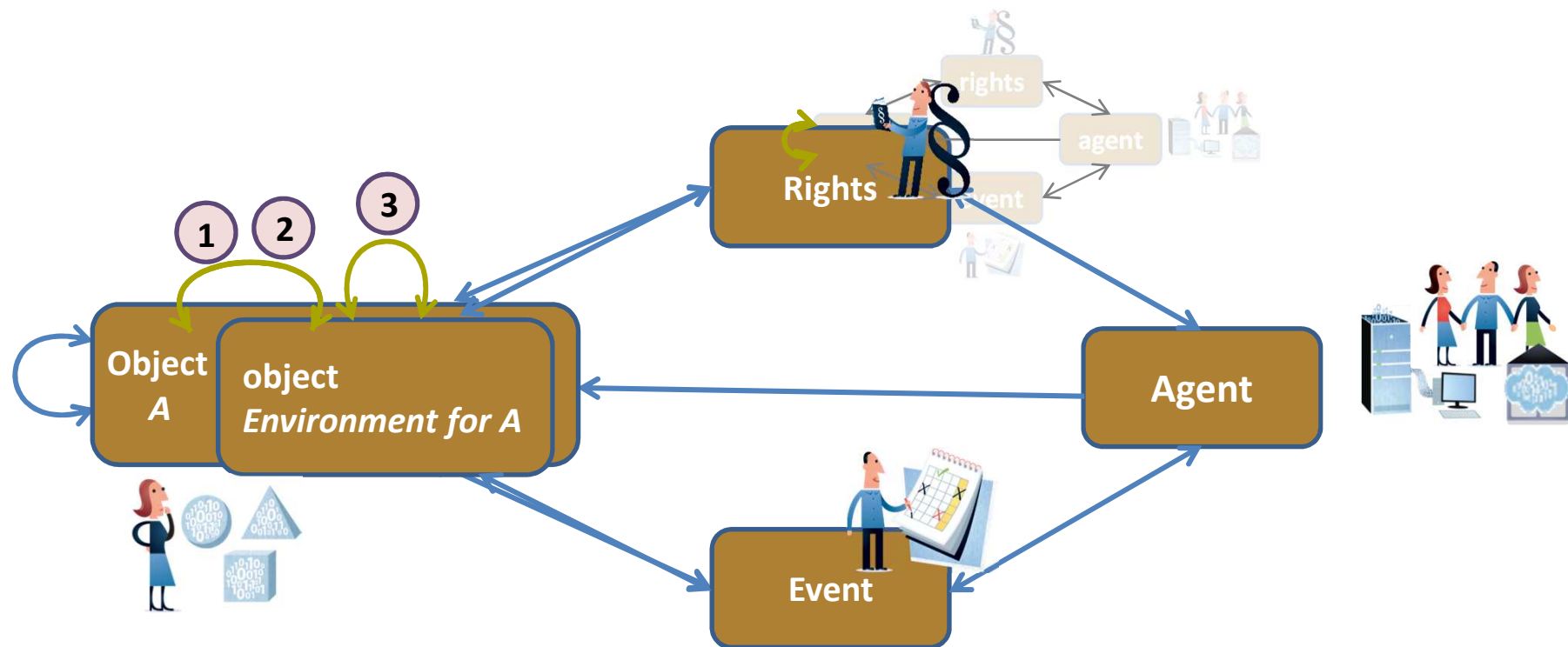


An agent is actually an environment acting as an agent

*e.g. a format migration software agent involved in a preservation action*

Relation points out the environment object acting as the agent





- |                               |   |
|-------------------------------|---|
| 1. object to environment      | - specify computational context             |
| 2. environment to object      | - documentation, specifications, surrogates |
| 3. environment to environment | - inclusion, dependency, derivation, other  |

## Additional environment information

- relationship  
Different environments can support different uses/purposes of objects  
**create, edit, modify, render...**
- relationship  
Characteristics describing how the environment supports its purpose  
**unspecified, minimum, known to work, recommended ...**

A controlled vocabulary is available at:

<http://id.loc.gov/vocabulary/preservation/environmentPurpose>

## PREMIS Object Entity – Semantic Units



relationship

relationshipType

relationshipSubType

relatedObjectIdentifier

(type, value, sequence)

relatedEventIdentifier

(type, value, sequence)

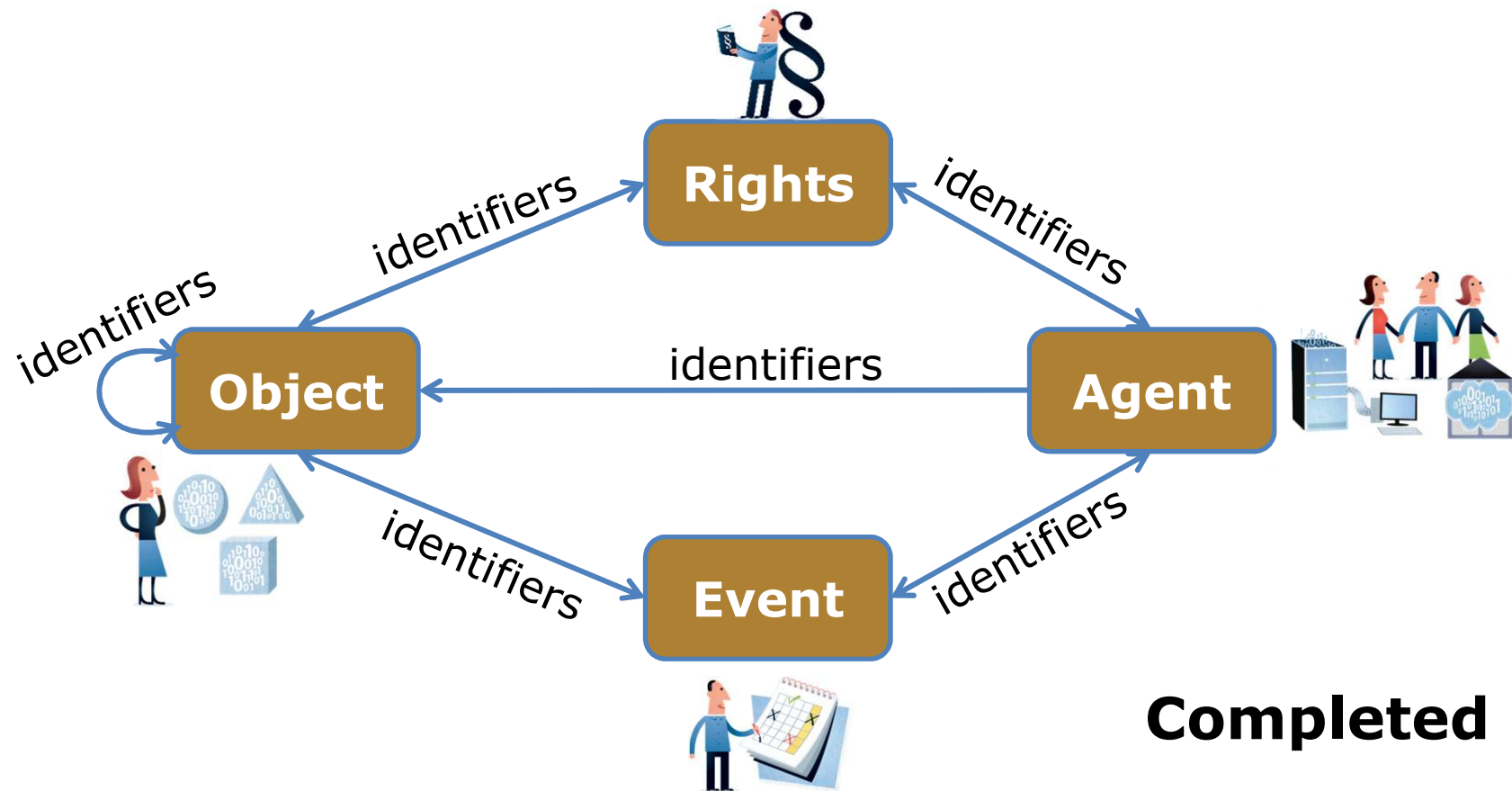
relatedEnvironmentPurpose

relatedEnvironmentCharacteristic

e.g. create, edit, modify, render

e.g. unspecified, minimum, recommended,  
known to work

## PREMIS 3 - Entities



Angela Di Iorio  
Sapienza Università di Roma



# HOW TO USE PREMIS?

The Data Dictionary in action:  
PREMIS expressions and  
conformance



## 1/ For building other standards

- As a basis extended with locally-defined elements: Preservation Metadata Dictionary (Netherlands Institute for Sound and Vision).[https://publications.beeldengeluid.nl/pub/389/BIIJLAGE-C\\_Metadatadictionary-English.pdf](https://publications.beeldengeluid.nl/pub/389/BIIJLAGE-C_Metadatadictionary-English.pdf)
- As a free source of inspiration: DEPIP (Data Exchange Protocol for Interoperability and Preservation), ISO 20614.

## 2/ As a self-assessment tool

- Am I able to provide information about my digital assets following the Data Dictionary structure and requirements?
  - I.e., documenting the mapping between my metadata structure and PREMIS semantic units.
- Conformance level 1 « through mapping »

### 3/ As an export format

- Preferably in a PREMIS-endorsed expression (XML or RDF)
- Conformance level 2 « through export »

## Serializing PREMIS

In XML:

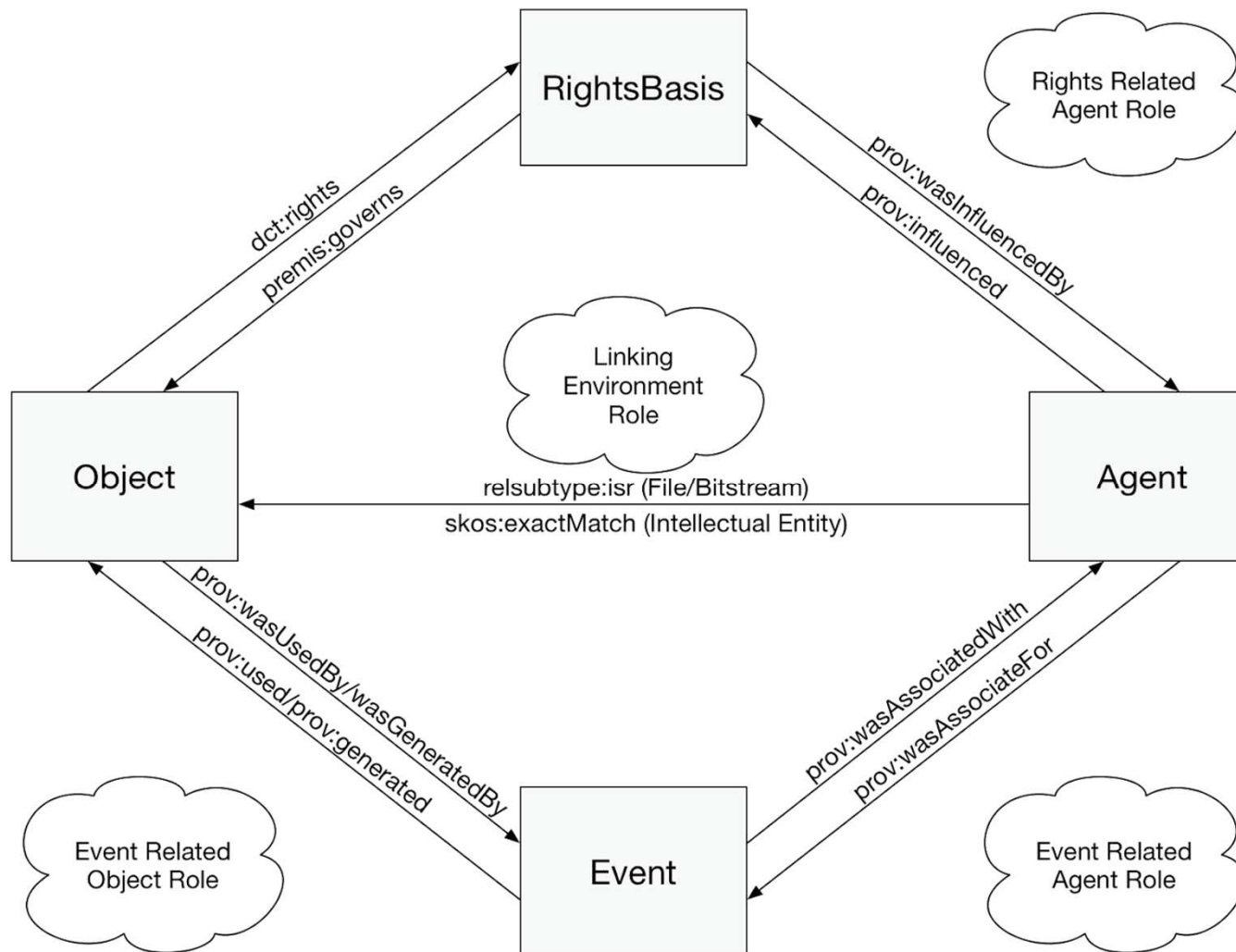
- Following strictly the DD structure: one semantic unit = one XML element
- Defined by one XSD schema

In RDF:

- Takes some liberties with the DD structure and semantic unit names to follow modelling best practices
- Defined by an OWL ontology compatible with preservation vocabularies defined at [id.loc.gov](http://id.loc.gov)

**Any other expression can be acceptable (YAML, JSON, etc.)!**

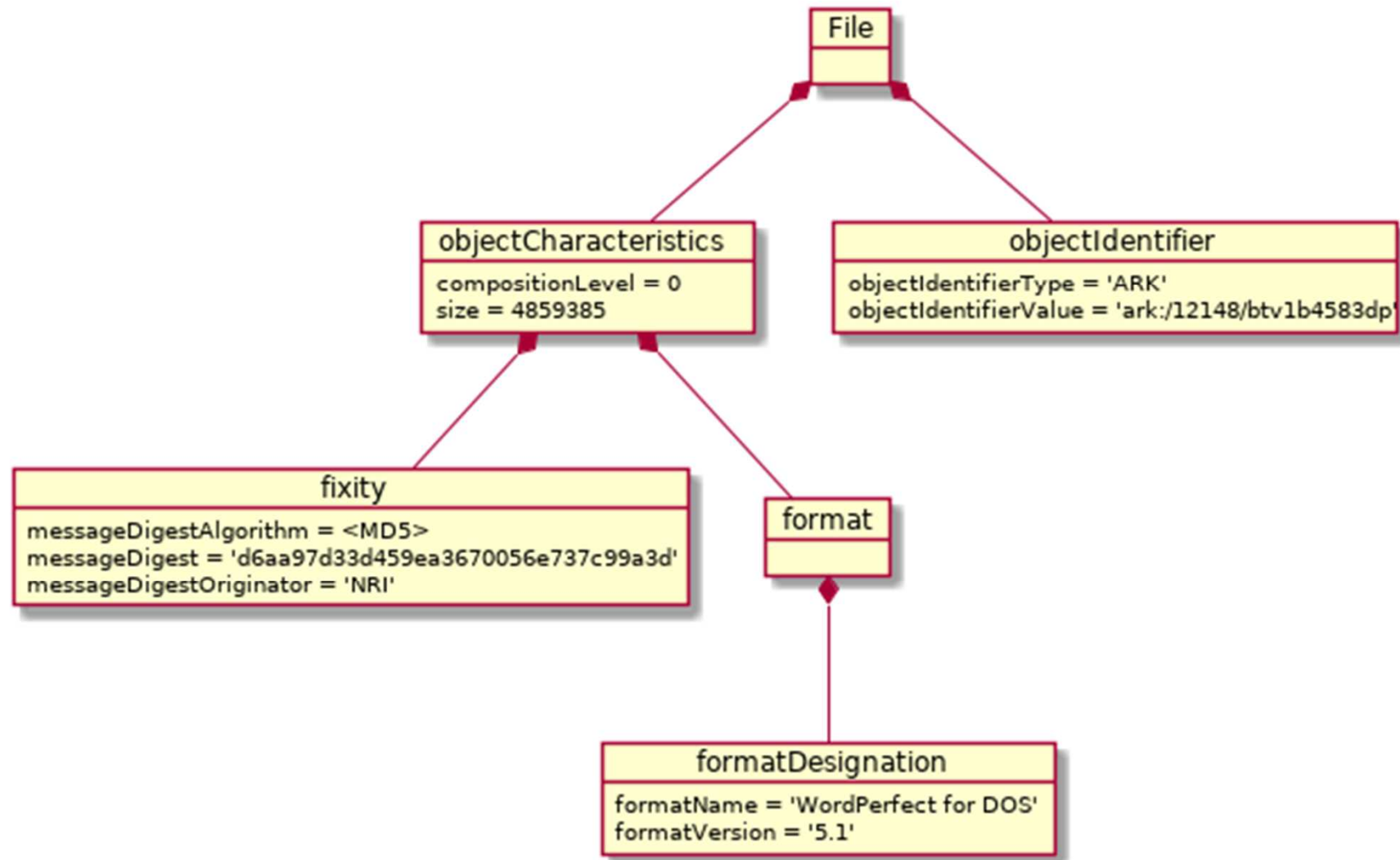
## The PREMIS Data Model, adapted to an RDF expression



**An example of both expressions with a simple statement...**

- There's a file called "ark:/12148/btv1b4583dp",
- Neither encrypted, nor compressed, nor packed,
- Whose MD5 checksum, calculated by a tool called "NRI", is "d6aa97d33d459ea3670056e737c99a3d",
- Whose size is 4,599,385 bytes,
- Which is a Wordperfect 5.1 file.

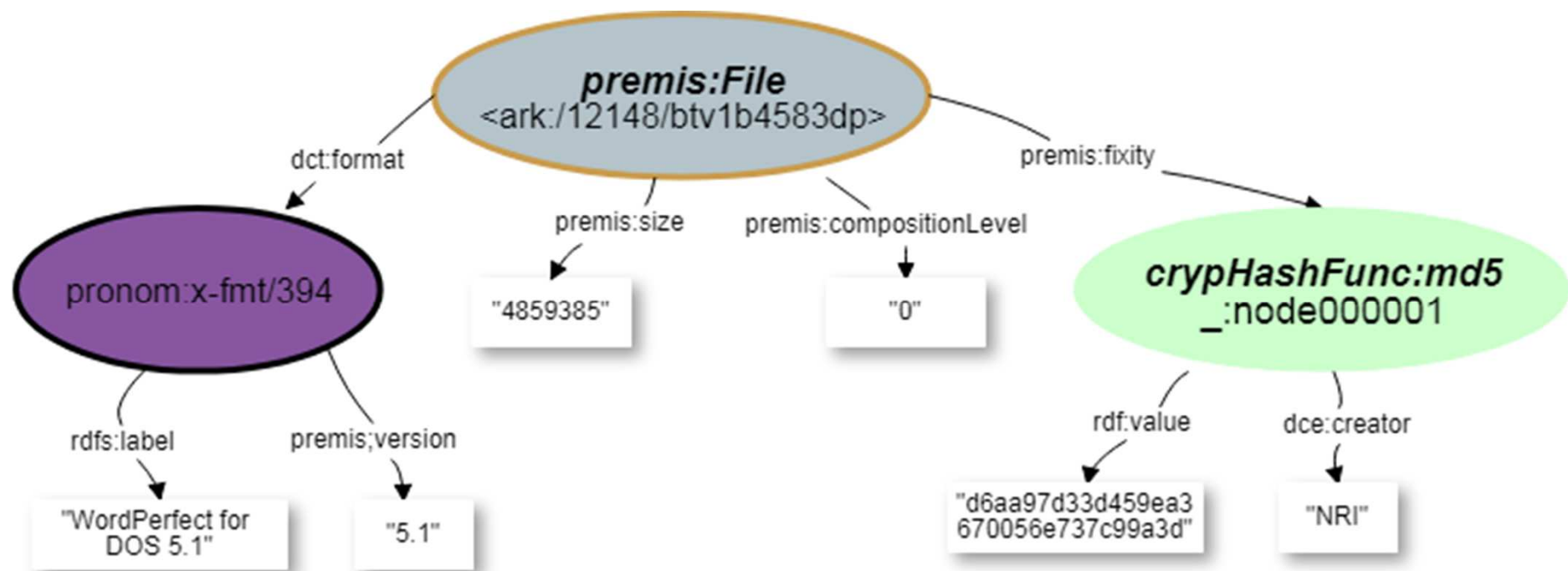
## Transcribed in PREMIS XML...



## Transcribed in PREMIS XML...

```
<?xml version="1.0" encoding="UTF-8"?>
<premis:object xsi:type="premis:file" xmlns:premis="http://www.loc.gov/premis/v3">
  <premis:objectIdentifier>
    <premis:objectIdentifierType>ARK</premis:objectIdentifierType>
    <premis:objectIdentifierValue>ark:/12148/btv1b4583dp</premis:objectIdentifierValue>
  </premis:objectIdentifier>
  <premis:objectCharacteristics>
    <premis:compositionLevel>0</premis:compositionLevel>
    <premis:fixity>
      <premis:messageDigestAlgorithm
valueURI="http://id.loc.gov/vocabulary/preservation/cryptographicHashFunctions/md5">MD5</
premis:messageDigestAlgorithm>
      <premis:messageDigest>d6aa97d33d459ea3670056e737c99a3d</premis:messageDigest>
      <premis:messageDigestOriginator>NRI</premis:messageDigestOriginator>
    </premis:fixity>
    <premis:size>4859385</premis:size>
    <premis:format>
      <premis:formatDesignation>
        <premis:formatName>WordPerfect for DOS</premis:formatName>
        <premis:formatVersion>5.1</premis:formatVersion>
      </premis:formatDesignation>
    </premis:format>
  </premis:objectCharacteristics>
```

## Or in PREMIS RDF



## Or in PREMIS RDF

```
<ark:/12148/btv1b4583dp> rdf:type premis:File ;
    premis:hasCompositionLevel "0" ;
    premis:hasFixity [ rdf:type
<http://id.loc.gov/vocabulary/preservation/cryptographicHashFunctions/md5> ;
        rdf:value "d6aa97d33d459ea3670056e737c99a3d" ;
        dce:creator "NRI" . ] ;
    premis:hasSize "4859385" ;
    dct:format
<https://www.nationalarchives.gov.uk/pronom/x-fmt/394> .

<https://www.nationalarchives.gov.uk/pronom/x-fmt/394>
rdfs:label "WordPerfect for DOS 5.1" ;
    premis:hasVersion "5.1" .
```

## 4/ As the native format of the repository Data Management module

Where to store PREMIS data?

- Any technology, using a PREMIS-endorsed expression or not, can be used
  - XML database
  - RDF triple store
  - relational database
  - etc.
  
- Conformance level 3 « through internal implementation »

## Which Entities to implement?

- Object is the core Entity (level A);
- Event and Agent are closely related (level B); implementing Agents has strong implications: it means the repository is able to manage and follow the use of its Agents in the Object lifecycle.
- The Rights Entity (excluded from the conformance statement) helps a repository tracking the intellectual property rights governing the Object, or some institutional policy.

## **PREMIS Conformance statement**

- <http://www.loc.gov/standards/premis/premis-conformance-20150429.pdf>

Angela Di Iorio  
Sapienza Università di Roma



# **PREMIS IMPLEMENTATION CASE STUDIES**

PREMIS with METS

Strategies for implementing

PREMIS with

Semantic Web Technology

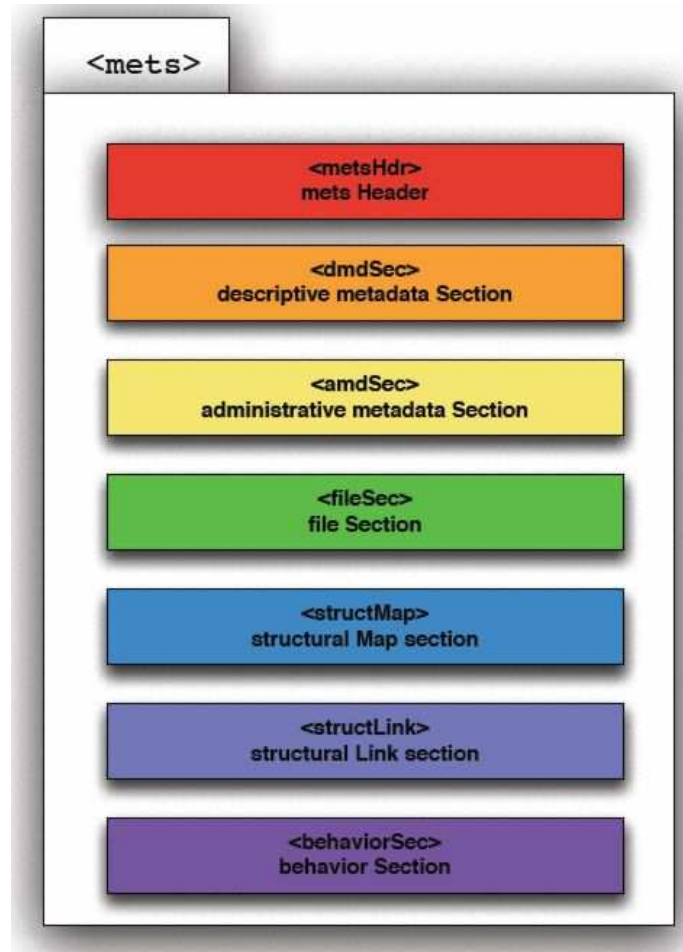


## PREMIS with METS

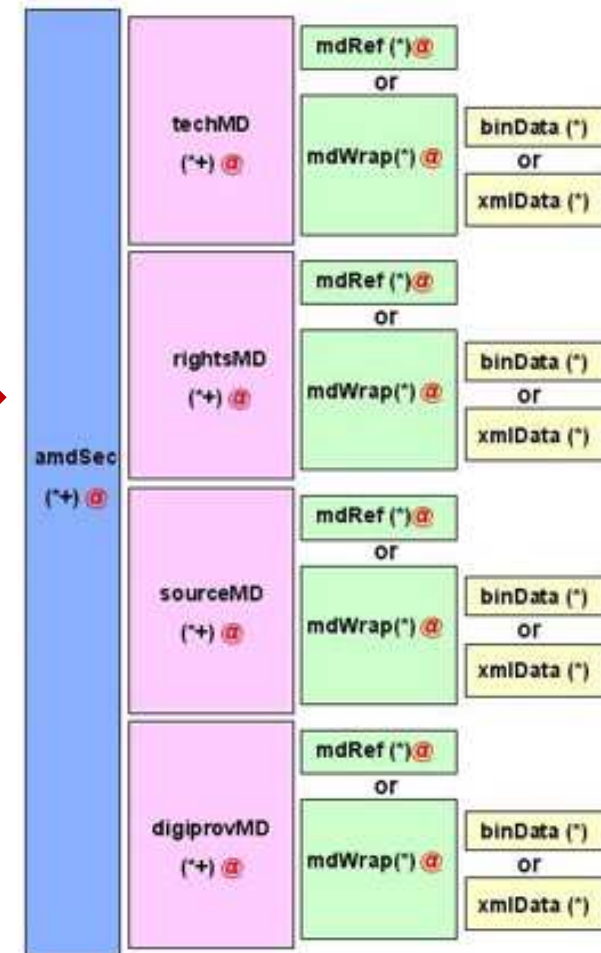
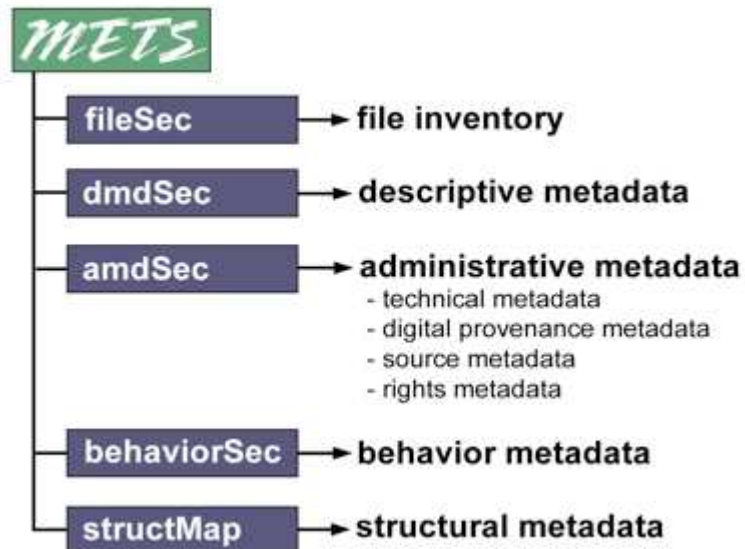
<http://www.loc.gov/standards/premis/guidelines2017-premismets.pdf>

- METS stands for Metadata Encoding Transmission Standard.
- It can be used for supporting different OAIS functional scenarios: Submission, Archival and Dissemination.
- Usually it is produced as an XML file that documents SIP, AIP or DIP.
- The content and structure of the exchanging OAIS IP can be conveyed by means of machine-readable information, that would support the exchange process between two information systems that need to exchange IPs.
- METS XML it is a PREMIS Object, which is essential in exchanging scenarios

## How is METS structured?



## How is METS structured?



<http://www.dlib.org/dlib/july08/guenther/07guenther.html>

## PREMIS with METS - distinct sections

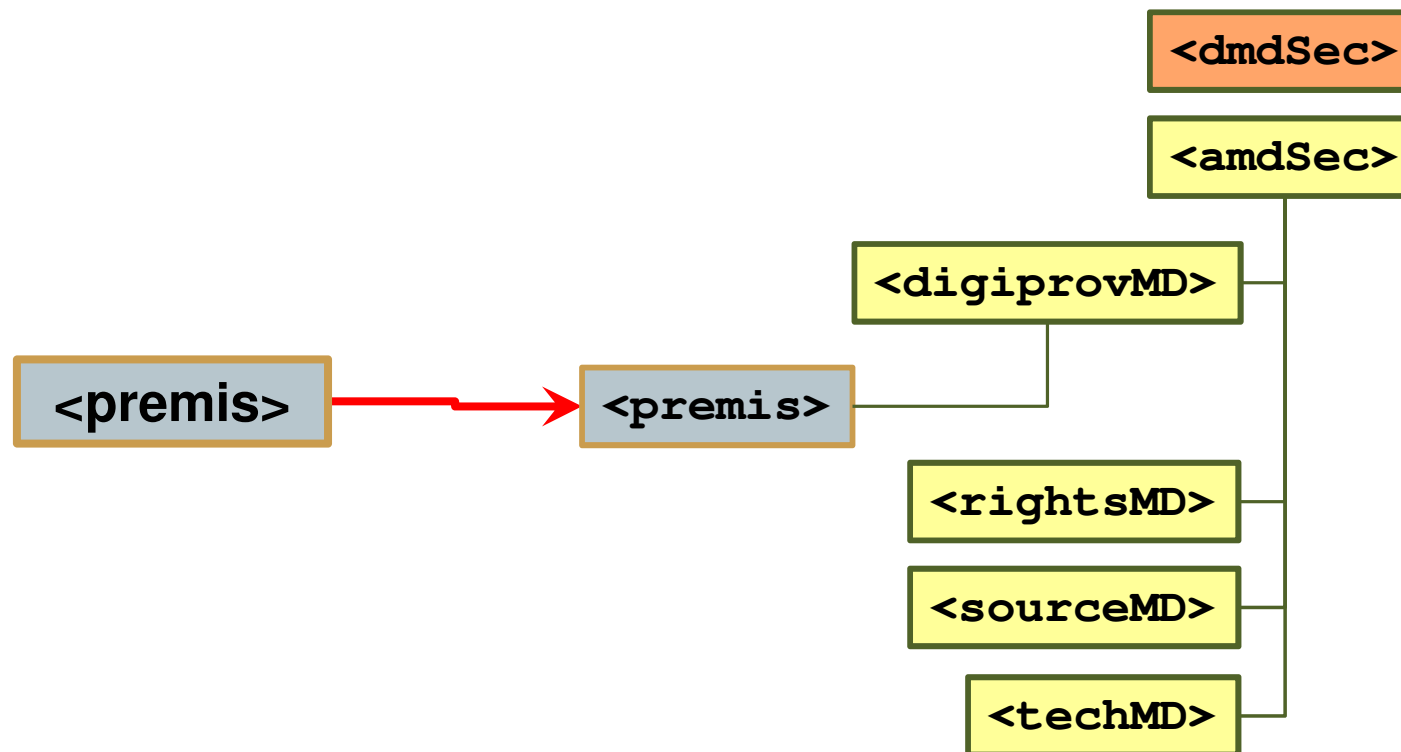
<http://www.loc.gov/standards/premis/guidelines2017-premismets.pdf>

PREMIS	METS
Representation File Bitstream Environment Intellectual Entity	mets:techMD
Intellectual Entity	mets:dmdSec
Event	mets:digiprovMD
Rights	mets:rightsMD
Agent-Event	mets:digiprovMD
Agent-Rights	mets:rightsMD

## PREMIS with METS – entire package

<http://www.loc.gov/standards/premis/guidelines2017-premismets.pdf>

PREMIS	METS
Object, Agent, Event, Rights	mets:digiprovMD/premis



## PREMIS with METS – redundancies

- Redundancies between PREMIS and METS

PREMIS	METS
objectCharacteristics/fixity objectCharacteristics/size objectCharacteristics/format  <pre> &lt;premis:objectCharacteristics&gt; &lt;premis:fixity&gt; &lt;premis:messageDigestAlgorithm&gt;MD 5&lt;/premis:messageDigestAlgorithm&gt; &lt;premis:messageDigest&gt;29e99f928d8 ec51720571a9a2259e57a&lt;/premis:mes sageDigest&gt; &lt;premis:messageDigestOriginator&gt;P HP 5.2.10 function&lt;/premis:messageDigestOri ginator&gt; &lt;/premis:fixity&gt; </pre>	<pre> &lt;mets:fileSec&gt; &lt;mets:fileGrp ID="JPEG-RMS45_00000001" VERSDATE="2013-12-16T19:02:59" USE="SOURCE"&gt; &lt;mets:file ID="FL-1" MIMETYPE="image/jpeg" ADMID="jpg-1" SIZE="2260142" CHECKSUM="6c78cb2a5953331d60f912a2601fa 03d" CHECKSUMTYPE="MD5" OWNERID="RMS45"&gt; &lt;mets:FLocat xlink:title="Cosmographia..." xlink:href="RMS45/RMS45_00000001/JPEG/R MS45_00000001_0001.jpg" LOCTYPE="URI"/&gt; &lt;/mets:file&gt; [...]</pre>

## PREMIS with METS – redundancies

Redundancies between PREMIS and other technical standards used in METS

PREMIS	MIX
<pre>&lt;premis:objectCharacteristics&gt; &lt;premis:fixity&gt; &lt;premis:messageDigestAlgorithm&gt;MD 5&lt;/premis:messageDigestAlgorithm&gt; &lt;premis:messageDigest&gt;29e99f928d8 ec51720571a9a2259e57a&lt;/premis:mes sageDigest&gt; &lt;premis:messageDigestOriginator&gt;P HP 5.2.10 function&lt;/premis:messageDigestOri ginator&gt; &lt;/premis:fixity&gt;</pre>	<pre>&lt;mix:Fixity&gt; &lt;mix:messageDigestAlgorithm&gt;MD5&lt;/m ix:messageDigestAlgorithm&gt; &lt;mix:messageDigest&gt;29e99f928d8ec51 720571a9a2259e57a&lt;/mix:messageDige st&gt; &lt;mix:messageDigestOriginator&gt;PHP 5.2.10 function&lt;/mix:messageDigestOrigina tor&gt; &lt;/mix:Fixity&gt;</pre>

## PREMIS with METS – structural relationships

PREMIS	METS structMap
relationship/relationshipType relationship/relationshipSubType <b>relationship/relatedObjectIdentifier</b> relationship/relatedObjectIdentifierType relationship/relatedObjectIdentifierValue relationship/relatedObjectSequence	<pre>&lt;mets:structMap TYPE="physical/logical" &gt; &lt;mets:div TYPE="book" DMDID="MODS- RMS45_00000001 DC-RMS45_00000001"&gt; &lt;mets:div LABEL="Cosmographia Petri Apiani"&gt; &lt;mets:div LABEL="Frontespizio" ORDER="1 "&gt; &lt;mets:fptr FILEID="FL-1"/&gt; &lt;mets:fptr FILEID="FL-139"/&gt; &lt;/mets:div&gt;</pre>

## PREMIS with METS – cross-referencing identifiers

METS fileSec	METS structMap
<pre>&lt;mets:fileSec&gt; &lt;mets:fileGrp ID="JPEG-RMS45_00000001" VERSDATE="2013-12-16T19:02:59" USE="SOURCE"&gt; &lt;mets:file ID="FL-1" MIMETYPE="image/jpeg" ADMID="jpg-1" SIZE="2260142" CHECKSUM="6c78cb2a5953331d60f912a2601fa03d " CHECKSUMTYPE="MD5" OWNERID="RMS45"&gt; &lt;mets:FLocat xlink:title="Cosmographia..." xlink:href="RMS45/RMS45_00000001/JPEG/RMS4 5_00000001_0001.jpg" LOCTYPE="URI"/&gt; &lt;/mets:file&gt; [...]</pre>	<pre>&lt;mets:structMap TYPE="physical/logica l"&gt; &lt;mets:div TYPE="book" DMDID="MODS- RMS45_00000001 DC-RMS45_00000001"&gt; &lt;mets:div LABEL="Cosmographia Petri Apiani"&gt; &lt;mets:div LABEL="Frontespizio" ORDER= "1"&gt; &lt;mets:fptr FILEID="FL-1"/&gt; &lt;mets:fptr FILEID="FL-139"/&gt; &lt;/mets:div&gt;</pre>

## PREMIS with METS – cross-referencing identifiers

METS fileSec	METS structMap
<b>Object/objectIdentifier</b> <b>Object/relationship/relatedObjectIdentifier</b> <b>Object/relationship/relatedEventIdentifier</b> <b>Object/linkingEventIdentifier</b> <b>Object/linkingRightsStatementIdentifier</b>  <b>Event/eventIdentifier</b> <b>Event/linkingAgentIdentifier</b> <b>Event/linkingObjectIdentifier</b>  <b>Agent/agentIdentifier</b> <b>Agent/linkingEventIdentifier</b> <b>Agent/linkingRightsStatementIdentifier</b> <b>Agent/linkingEnvironmentIdentifier</b>  <b>Rights/rightsStatement/rightsStatementIdentifier</b> copyrightDocumentationIdentifier licenseDocumentationIdentifier statuteDocumentationIdentifier otherRightsDocumentationIdentifier  <b>Rights/linkingObjectIdentifier</b> <b>Rights/linkingAgentIdentifier</b>	<pre> &lt;mets:structMap TYPE="physical/logical"&gt;   &lt;mets:div TYPE="book" DMDID="MODS- RMS45_00000001 DC-RMS45_00000001"&gt;     &lt;mets:div LABEL="Cosmographia Petri Apiani"&gt;       &lt;mets:div LABEL="Frontespizio" ORDER= "1"&gt;         &lt;mets:fptr FILEID="FL-1"/&gt;         &lt;mets:fptr FILEID="FL-139"/&gt;       &lt;/mets:div&gt;     &lt;/mets:div&gt;   &lt;/mets:div&gt; </pre>



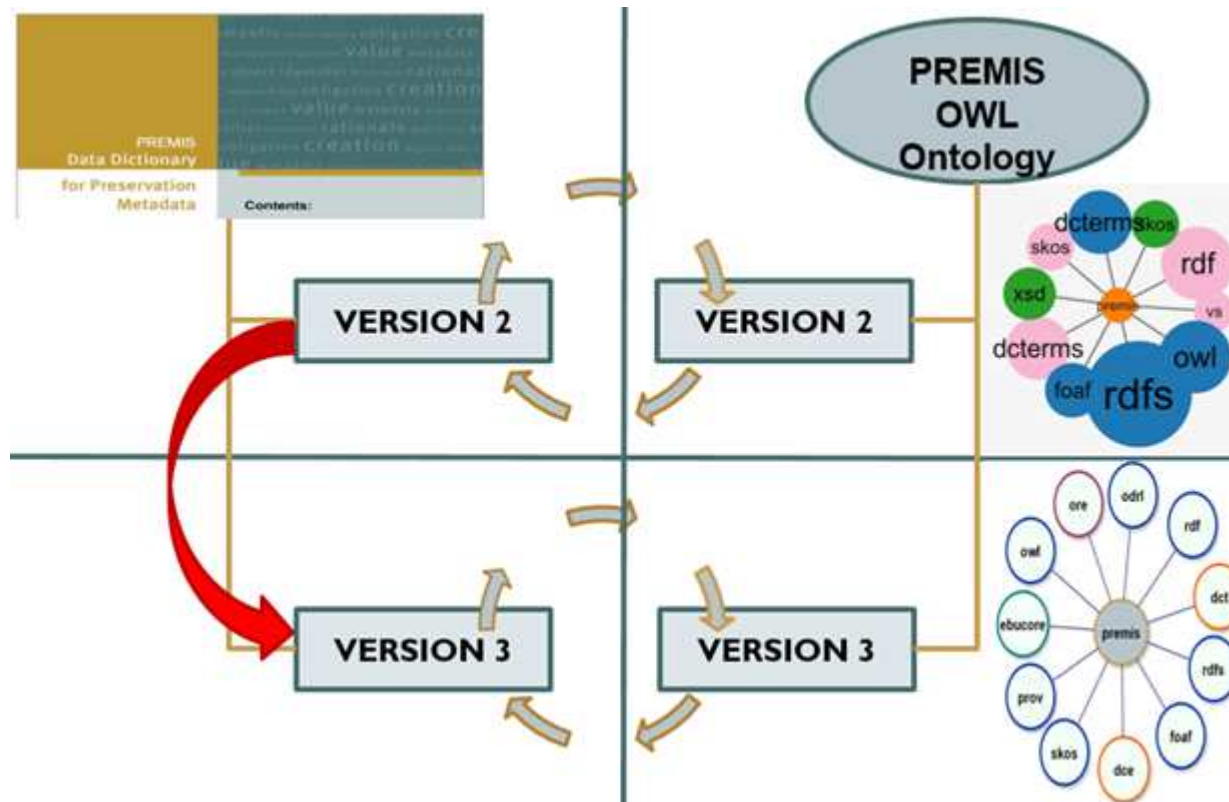
## Using PREMIS with METS – web page

<http://www.loc.gov/standards/premis/premis-mets.html>

## Strategies for implementing PREMIS with Semantic Web Technology

- **Semantic Web Technology?**
  - Schemes for web languages (**RDFS, OWL**),
  - knowledge classification (**SKOS**)
  - and data framework (**RDF**)
  
  - A set of best practices (Linked Data)

### PREMIS supports the implementation of Semantic Web Technology



**SKOS**

**LIBRARY** LIBRARY OF CONGRESS

The Library of Congress > Linked Data Service

ID.LOC.GOV

ID.LOC.GOV – Linked Data Service  
ID.LOC.GOV provides both interactive and machine  
bibliographic description. To search everything, use  
vocabulary.

**LINKED DATA**  
principles and  
best practices

## Linked Data principles

### 1) Use URIs as names for things

“not just Web documents and digital content, but also real world objects and abstract concepts”.

### 2) Use HTTP URIs so that people can look up those names

“to identify objects and abstract concepts”.

### 3) When someone looks up a URI, provide useful information, using the standards (RDF\*, SPARQL)

“use of a single data model for publishing structured data on the Web a simple graph-based data model that has been designed for use in the context of the Web”.

### 4) Include links to other URIs, so that they can discover more things

“not only Web documents, but any type of thing”.

## SKOS Simple Knowledge Organization System

1) thesauri, taxonomies, classification schemes, etc., expressed in SKOS RDF are machine-readable and, software applications can use them as a Semantic Web knowledge;

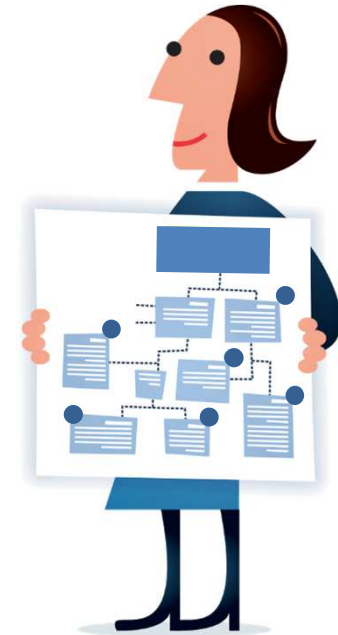
2) lightweight, intuitive conceptual modeling language for developing and sharing controlled vocabularies in the Semantic Web;

3) a bridging technology between the rigorous logical formalism of OWL, and the informal and weakly-structured data.

**Karin Bredenberg**  
National Archives of Sweden



## WRAP UP



- DD
- Where to find it and What it is for
- Current activity in PREMIS EC
- Exercises
- Book



**Data Dictionary  
for Prevention  
Abundant**

David A. G. Reardon

**Contents**

- Introduction
- The Data Dictionary
- The Data Dictionary: A Review of the Literature
- Appendix
- Index
- References
- Index

**CRC**  
Taylor & Francis Group

For more information on this book, please contact:  
CRC Press, 2001 N. Zeeb Road, Boca Raton, FL 33431  
Tel: 888-451-4243 or 561-845-4243  
Fax: 561-845-4243  
Email: [info@crcpress.com](mailto:info@crcpress.com)

PREMIS Tutorial at iPres2

Semantic	Entity set	CONTENTS	
		Acknowledgments	iii
		PREMIS Editorial Committee members	v
		Special thanks	v
		PREMIS Web Sites and E-Mail	viii
		Introduction	1
		Background	1
		Development of the original PREMIS Data Dictionary	1
		Implementable, core preservation metadata	2
		PREMIS Maintenance Activity	3
		Version History	4
		PREMIS Awards and Recognition	5
		The PREMIS Data Model	6
		More on Objects	8
		More on Events	15
		More on Agents	16
		More on Rights	17
		General Topics on the Structure and Use of the Data Dictionary	17
		Identifiers	17
		Relationships between Objects	19
		Relationships between entities of different types	21
		The 1:1 principle	21
		Implementation Considerations	22
		PREMIS conformance	22
		Implementation of the data model	24
		Storing metadata	25
		Supplying metadata values	25
		Extensibility	27
		Date and time formats in PREMIS	29
		The PREMIS Data Dictionary Version 3.0	30
		Limits to the scope of the Data Dictionary	31
		Object Entity	33
		Entity types	33

## Where? and What??

- Where
  - Resources: <http://www.loc.gov/standards/premis/>
  - PREMIS Implementors Group Forum:  
[PIG@listserv.loc.gov](mailto:PIG@listserv.loc.gov)
- What PREMIS is for today have given you a good explanation to and you now need to explore it further



Images in this style is taken from [digitalbevaring.dk](http://digitalbevaring.dk)

## Resources

- Understanding PREMIS
- PREMIS-in-METS guidelines
- Conformance statement
- Examples of implementation

<http://www.loc.gov/standards/premis/>

## Current activity

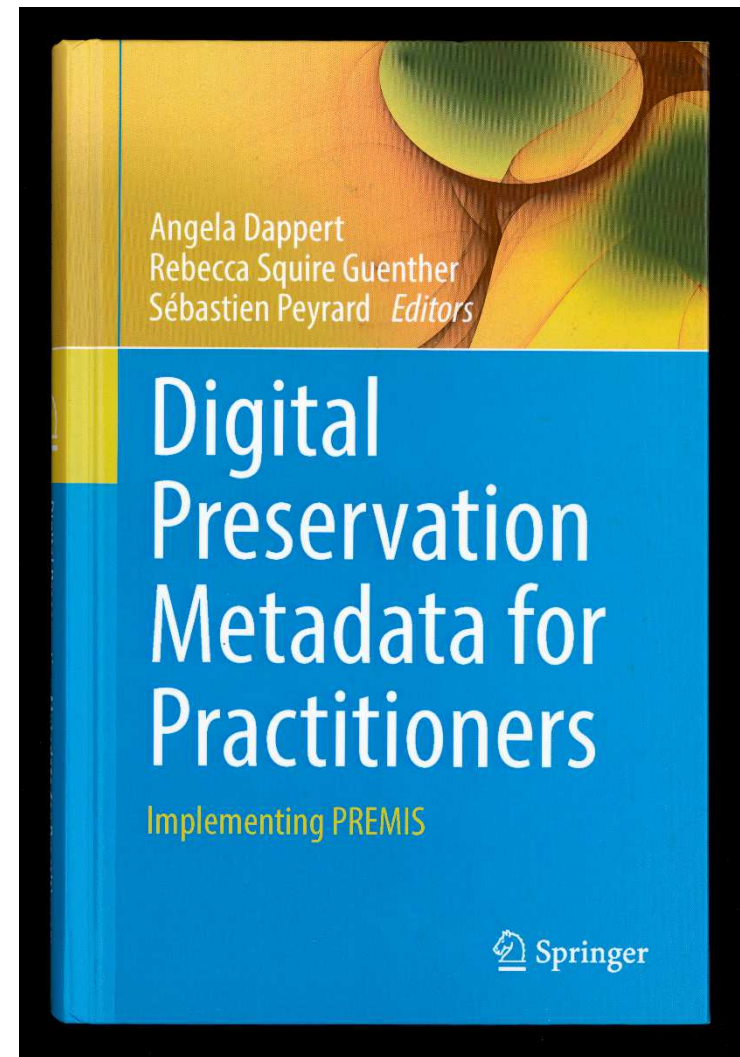
- Move the DD to a TEI-format to simplify maintenance and transformations to publications
- DD updates following the ontology work
- Rights overhaul

## Exercises

- Today have been really filled!
- Three exercises to start working with PREMIS metadata
  - Print them out!
- Solutions is also published!
- <http://www.loc.gov/standards/premis/pif/2019/premis-implementation-fair-announcement-2019.html>

## Book

- ISBN E-book:  
978-3-319-43763-7
- ISBN Hardcover:  
978-3-319-43761-3
- <http://www.springer.com/gp/book/9783319437613>



## Finally...

PREMIS is a community standard.

- Send examples
- Ask questions
- Send suggestions
- Take part!

**Thank you!**

Angela, Bertrand, Eld and Karin